MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOULOUD MAMMERI, TIZI-OUZOU

FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT D'INFORMATIQUE

MEMOIRE DE MASTER

Présenté par :

DOUNAS Tarik

OULD FELLA Makhlouf

En vue de l'obtention du diplôme de Master en informatique

Option : Ingénierie des Systèmes d'Information

Intitulé :

Implémentation d'un modèle d'appariement pour un Système de Recherche d'Information Personnalisé

Devant le jury d'examen composé de :

Mme. FELLAG	**Présidente**
Mme. ACHEMOUKH	**Promotrice**
Mme. HADDAOUI	**Examinatrice**
Mme. AIT ADDA	**Examinatrice**

Soutenu le : 22 /10/ 2015

Remerciements :

Nous remercions en premier lieu le Bon Dieu tout puissant qui nous a donnés le courage et la patience pour réaliser ce mémoire.

Arrivée au terme de notre travail, nous tenons à exprimer notre gratitude à notre promotrice Mme ACHEMOUKH qui nous a dirigées tout au long de notre travail,

Nos chaleureux remerciements et gratitude vont à nos chers enseignants de notre spécialité informatique de nous avoir transmis leurs connaissances et leur savoir,

Enfin, Tous nos remerciements aux membres du jury qui nos feront l'honneur de juger notre travail.

Par ailleurs, nous remercions tous ceux qui nos ont aidés de loin ou de prêt à la réalisation de ce modeste travail.

Dédicace

Nous dédions ce modeste travail A nos parant,

Nos familles, nos camarades de notre promotion

Nos proches et tous nos amis (es)

Et enfin toute personnes ayant contribue au bon accomplissement de notre projet.

Sommaire

Chapitre II : **la recherche d'information personnalisée**

Liste des figures

Introduction Générale

Introduction générale

Aujourd'hui, l'information joue un rôle primordial dans le quotidien des individus et dans l'essor des entreprises. Cependant, le développement de l'Internet et la généralisation de l'informatique dans tous les domaines ont conduit à la production d'un volume d'information sans précédent. En effet, la quantité d'information disponible, particulièrement à travers le web, se mesure en milliards de pages. Il est par conséquent, de plus en plus difficile de localiser précisément ce que l'on recherche dans cette masse d'information.

La recherche d'information (RI) est le domaine par excellence qui s'intéresse à répondre à ce type d'attente. En effet, l'objectif principal de la RI est de fournir des modèles, des techniques et des outils pour stocker et organiser des masses d'informations et localiser celles qui seraient pertinentes relativement à un besoin en information d'un utilisateur, souvent, exprimé à travers une requête. Ces outils sont appelés des Systèmes de Recherche d'Information (SRI).

L'objectif d'un système de recherche d'information (SRI) est de faciliter l'accès à un ensemble de documents, afin de permettre à l'utilisateur de retrouver ceux qui sont pertinents, c'est-à-dire ceux dont le contenu correspond le mieux à son besoin en information.

Les systèmes de recherche d'information classiques se basent sur une recherche par mots clés, les documents sont représentés comme des sacs de mots et la pertinence d'un document vis-à-vis d'une requête est souvent estimée en s'appuyant sur les fréquences d'apparition des mots de la requête dans ces mêmes documents sans donner d'importance à la profession de l'utilisateur par exemple deux utilisateurs de deux domaines différents tape la même requête, les résultats retournés par le système de recherche d'informations serons les même.

Pour remédier à ce problème, de nouvelles approches ont été développés dans le but d'intégrer l'utilisateur dans l'une des phases de processus de recherche afin de personnaliser le SRI.

Notre travail s'inscrit principalement dans ce contexte. Notre objectif est alors d'implémenter un modèle d'appariement pour une recherche d'information personnalisée.

Pour mieux cerner l'objectif de notre travail, dans le premier chapitre nous introduisons la recherche d'information classique et le processus de recherche d'information. On présente par la suite les différents modèles de la RI et les principaux paramètres d'évaluation d'un SRI.

Dans le deuxième chapitre nous détaillerons la recherche d'information personnalisée et on expliquera les déférentes approches et techniques de modélisation du profil, ainsi que son intégration dans les déférentes phases du processus de recherche, et on donnera l'architecture générale d'un SRI Personnalisé. Le troisième chapitre concerne la modélisation du profil utilisateur et son intégration dans la phase d'appariement du processus de recherche d'information. Et le quatrième chapitre porte sur la mise en œuvre de notre système de recherche d'information personnalisé. Enfin, nous terminerons notre mémoire par une conclusion générale et les perspectives envisagées.

Chapitre I : La recherche d'information classique

I.1. Introduction

La Recherche d'Information (RI) n'est pas un domaine récent, il date des années 40. Une des premières définitions de la RI a été donnée par Salton : « la recherche d'information est un domaine qui a pour objectif, la représentation, l'analyse, l'organisation, le stockage et l'accès à l'information » **[Salton, 1984]**.

Plusieurs tâches se regroupent sous le vocable de la RI, la plus ancienne est la recherche documentaire, on y trouve également d'autres tâches plus au moins récentes comme : le filtrage d'information, l'extraction d'information, la recherche d'information multilingue, les questions réponses, la recherche d'information sur le web, etc.

Ce chapitre a pour but de présenter le domaine de la RI. Dans la première partie, nous présentons les concepts de base de la RI. En particulier, nous décrivons les notions de document, de requête et de pertinence ; les processus d'indexation, de recherche et de reformulation de requêtes ; ainsi que, les modèles de RI. Dans la dernière partie de ce chapitre est discutée l'évaluation des systèmes de recherche d'information.

I.2. Concepts de base de la recherche d'information

Le rôle d'un Système de Recherche d'Information (SRI) est de mettre en œuvre des techniques et des moyens permettant de retourner les documents pertinents d'une collection en réponse à un besoin en information d'un utilisateur, exprimé par un langage de requêtes qui peut être le langage naturel, une liste de mots clés ou un langage booléen**[Ribeiro-Neto, 2011]**.

Afin d'atteindre cet objectif, un processus d'indexation des documents de la collection est effectué. Il permet de construire une représentation synthétique des documents, appelée index.

Lorsque l'utilisateur formule sa requête un processus similaire est effectué sur la requête. Il consiste à analyser la requête et établir une représentation interne. Puis, le système établit une correspondance entre la représentation de la requête et la représentation des documents (index) pour sélectionner et présenter les documents qui répondent le mieux au besoin en information de l'utilisateur (les documents pertinents).

Le SRI s'appuie sur des modèles de RI pour établir cette correspondance entre les documents et la requête.

L'architecture générale d'un SRI illustrée par la figure I.1 fait ressortir des éléments constitutifs tels que : le document, le besoin en information, la requête et la pertinence, ainsi

que trois principales fonctionnalités : l'indexation, la recherche et la reformulation de la requête.

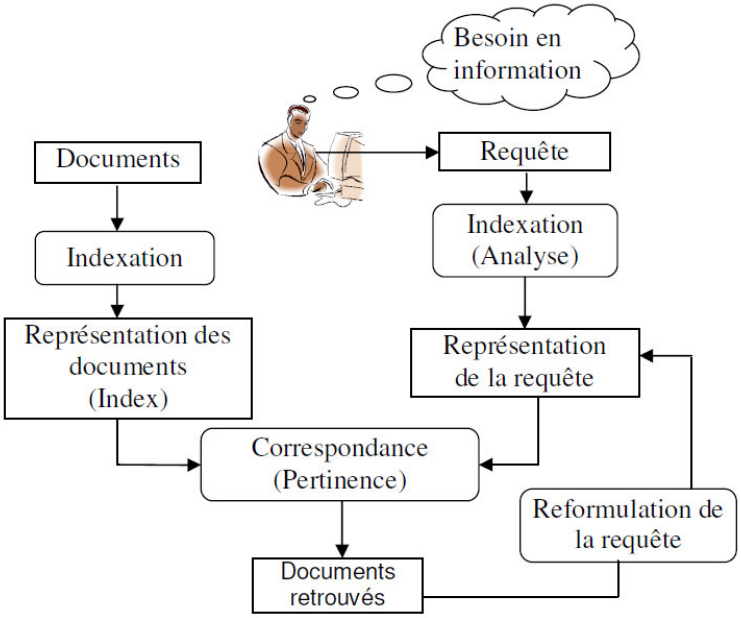

Figure I.1 Architecture générale d'un SRI

Document et collection de documents

Un document est un élément essentiel dans un SRI. Dans son acceptation courante, l'une des définitions possible du terme document est de le considérer comme un support physique de l'information, qui peut être du texte, une page web, une image, etc.

Dans le cas d'un document texte on peut le représenter selon trois vues **[Salton, 1983]** :

La vue sémantique (ou contenu) : elle se concentre sur l'information véhiculée dans le document.

La vue logique : elle définit la structure logique du document (structuration en chapitres, sections)

La vue présentation : elle consiste en la présentation sur un médium à deux dimensions (alignement de paragraphes, indentation, en-têtes et pieds de pages, etc.).

L'ensemble des documents manipulés par un SRI se nomme collection de documents.

Besoin en information et requête

La requête est une expression approximative du besoin en information de l'utilisateur. Ce dernier est une expression mentale des informations que l'utilisateur recherche.

Les requêtes soumises au SRI par les utilisateurs peuvent ne pas refléter leurs besoins en information. Cela est dû, d'une part, au fait que l'utilisateur ignore le fonctionnement interne du SRI, et il n'a qu'une vision restreinte des documents disponibles dans la collection.

D'autre part, le SRI n'a souvent aucune connaissance a priori de ses utilisateurs (centres d'intérêts, niveaux, parcours, etc.). Ce biais entre la requête et le besoin en information est une des difficultés majeures de tout système de recherche d'information.

Afin de remédier partiellement à ce problème un mécanisme de reformulation de requêtes peut être intégré dans les SRI.

Pertinence

La pertinence est une notion fondamentale et cruciale dans le domaine de la RI. Cependant, la définition de cette notion complexe n'est pas simple, car elle fait intervenir plusieurs notions **[mizzaro, 1997].** Basiquement, elle peut être définie comme la correspondance entre un document et une requête ou encore comme une mesure d'informativité du document à la requête.

Essentiellement, deux types de pertinence sont définis : la pertinence système et la pertinence utilisateur.

La pertinence Système est souvent présentée par un score attribué par le SRI afin dévaluer l'adéquation du contenu des documents vis-à-vis de celui de la requête. Ce type de pertinence est objectif et déterministe **[Cleverdon, 1970]**.

Pertinence utilisateur quant à elle, se traduit par les jugements de pertinence de l'utilisateur sur les documents fournis par le SRI en réponse à une requête. La pertinence utilisateur est subjective, car pour un même document retourné en réponse à une même requête, il peut être jugé différemment par deux utilisateurs distincts (qui ont des centres d'intérêt différents). De plus, cette pertinence est évolutive, un document jugé non pertinent à l'instant t pour une requête peut être jugé pertinent à l'instant $t+1$, car la connaissance de l'utilisateur sur le sujet a évolué **[Harter, 1992]**.

I.2.1 Le processus d'indexation

Pour que la recherche d'information se réalise avec des coûts acceptables, il convient d'effectuer une opération fondamentale sur les documents de la collection. Cette opération est nommée indexation **[Raghavan, 2008]**. Elle consiste à associer à chaque document une liste de mots clés appelée aussi descripteur, susceptible de représenter au mieux le contenu sémantique des documents.

La finalité de l'indexation est donc de produire une représentation synthétique des documents, formé de termes, ces termes peuvent être extraits de trois manières

Manuelle : chaque document de la collection est analysé par un spécialiste du domaine ou un documentaliste. L'indexation manuelle assure une meilleure précision dans les documents restitués par le SRI en réponse aux requêtes des utilisateurs **[Ren, 1999]**.

Néanmoins, cette indexation présente un certain nombre d'inconvénients liés notamment à l'effort et le prix qu'elle exige (en temps et en nombres de personnes).

Semi-automatique : la tâche d'indexation est réalisée ici conjointement par un programme informatique et un spécialiste du domaine **[Jacquemin, 2002]**. Le choix final des descripteurs revient à l'indexeur humain. Dans ce type d'indexation un langage d'indexation contrôlé est généralement utilisé.

Automatique : dans ce cas, l'indexation est entièrement automatisée. Elle est réalisée par un programme informatique et elle passe par un ensemble d'étapes pour créer d'une façon automatique l'index. Ces étapes sont : l'analyse lexicale, l'élimination des mots vides, la normalisation (lemmatisation ou radicalisation), la sélection des descripteurs, le calcul de statistiques sur les descripteurs et les documents (fréquence d'apparition d'un descripteur dans un document et dans la collection, la taille de chaque document, etc.) et enfin la création de l'index et éventuellement sa compression. Nous détaillons ces différentes étapes ci-dessous :

I.2.1.1 L'analyse lexicale

Elle permet de convertir un texte de document en une liste de termes. Un terme est un groupe de caractères constituant un mot significatif **[Fox, 1992]**. L'analyse lexicale permet de reconnaître les espaces de séparation des mots, les chiffres, les ponctuations, etc.

I.2.1.2 L'élimination des mots vides

Les mots vides (article, proposition, conjonction, etc.) sont des mots non significatifs dans un document, car ils ne traitent pas le sujet du document.

On distingue deux techniques pour éliminer les mots vides :

- L'utilisation d'une liste préétablie de mots vides (aussi appelée *anti dictionnaire* ou *stop-list*),

- L'élimination des mots ayant une fréquence qui dépasse un certain seuil dans la collection. L'élimination des mots vides réduit la taille de l'index, ce qui améliore le temps de réponse du système.

I.2.1.3 La normalisation

La normalisation consiste à représenter les différentes variantes d'un terme par un format unique appelé lemme ou racine. Ce qui a pour effet de réduire la taille de l'index. Plusieurs stratégies de normalisation sont utilisées : la table de correspondance, l'élimination des affixes(l'algorithme de Porter **[Porter, 1993]**), la troncature, l'utilisation des N-grammes **[Adamson, 1974]**.

L'inconvénient majeur de cette opération est qu'elle supprime dans certains cas la sémantique des termes originaux, c'est le cas par exemple des termes derivate/derive, activate/active, normalisés par l'algorithme de Porter.

I.2.1.4 Le choix des descripteurs

Elle consiste à déterminer le type d'unités élémentaires pour représenter les documents. On parle aussi de descripteur. L'objectif est d'avoir une représentation des documents permettant une moindre perte d'information sémantique possible. On distingue plusieurs types de descripteurs**[Baziz, 2005]**.

- **Les mots simples** : les mots simples du texte de document en éliminant les mots vides,

- **Les lemmes** ou les racines des mots extraits.

- **Les N-grammes** : qui sont une représentation originale d'un texte en séquence de N caractères consécutifs. On trouve des utilisations de bi-grammes et trigrammes dans la recherche d'information.

- **Les mots composés :** groupes de mots ou expression (phrase en anglais) sont souvent plus riches sémantiquement que les mots qui les composent pris séparément.

- **Les concepts :** qui sont des expressions pris généralement d'une structure conceptuelle, tels que les thésaurus ou les ontologies.

I.2.1.5 La création de l'index

Au terme du processus d'indexation, un ensemble de structure de données sont crées. Ces dernières permettent un accès efficace à la représentation des documents. Le fichier inverse est la structure de données la plus utilisée **[Manning, 2008]**, il enregistre pour chaque descripteur les identificateurs des documents qui le contiennent et sa fréquence dans chacun de ces documents.

Généralement, les structures de données sont compressées avant d'être enregistrées sur le disque, ce qui permet de réduire la taille de l'index.

I.2.2 Appariement document-requête

La fonction d'appariement document-requête permet de mesurer la valeur de pertinence d'un document vis-à-vis d'une requête. Afin de réaliser cela, le SRI représente le document et la requête avec un même formalisme, puis le SRI compare les deux représentations. Le résultat de cette comparaison se traduit par un score qui détermine la probabilité de pertinence (degré de similarité ou degré de ressemblance) du document vis-à-vis de la requête. Cette fonction d'appariement est notée *SRV* (*d,q*). (*R*etrieval *S*tatut *V*alue), où ***d*** représente un document de la collection et ***q*** la requête. Cette valeur permet ensuite au SRI d'ordonner les documents renvoyés à l'utilisateur.

I.2.3 Les modèles de recherche d'information

Un modèle de RI fournit une interprétation théorique de la notion de pertinence. Plusieurs modèles de RI on été proposés dans la littérature, ils s'appuient sur des cadres théoriques différents, théorie des ensembles, algèbre, probabilités, etc. Globalement, on distingue trois principales catégories de modèles: modèles booléens, modèles vectoriels et modèles probabilistes **[Dominich, 2001]**.

I.2.3.1 Le modèle booléen

Les premiers SRI développés sont basés sur le modèle booléen, même aujourd'hui beaucoup de systèmes commerciaux (moteurs de recherche) utilisent le modèle booléen. Cela est dû à la simplicité et à la rapidité de sa mise en œuvre.

Le modèle booléen est basé sur la théorie des ensembles et l'algèbre de Boole.

Dans ce modèle, un document ***d*** est représenté par un ensemble de mots-clés (termes) ou encore un vecteur booléen. La requête ***q*** de l'utilisateur est représentée par une expression logique, composée de termes reliés par des opérateurs logiques : ET (Λ), OU (v) et SAUF (¬).

L'appariement (*RSV*) entre une requête et un document est un appariement exact, autrement dit si un document implique au sens logique la requête alors le document est pertinent. Sinon, il est considéré non pertinent. La correspondance entre document et requête est déterminée comme suit :

$$RSV\,(d,\,q) = \begin{cases} 1 \text{ si } d \text{ appartient a l'ensemble décrit par } q \\ 0 \text{ si non} \end{cases}$$

Malgré la large utilisation de ce modèle, il présente un certain nombre de faiblesses :

- Les documents retournés à l'utilisateur ne sont pas ordonnés selon leur pertinence.

- La représentation binaire d'un terme dans un document est peu informative, car elle ne renseigne ni sur la fréquence du terme dans le document ni sur la longueur de document, qui peuvent constituer des informations importantes pour la RI.

- Ce modèle ne supporte pas la réinjection de pertinence.

Afin de remédier à certains problèmes de ce modèle, des extensions ont été proposées, parmi elles on trouve : le modèle booléen basé sur la théorie des ensembles flous **[Radecki, 1979]**, le modèle booléen étendu **[Salton, 1983]**.

I.2.3.2 Le modèle vectoriel

Le modèle vectoriel de base a été introduit par Salton **[Salton, 1971]**, concrétisé dans le cadre du système SMART. Ce modèle se base sur une formalisation géométrique. En effet, les documents et les requêtes sont représentés dans un même espace, défini par un ensemble de dimensions, chaque dimension représente un terme d'indexation.

Les requêtes et les documents sont alors représentés par des vecteurs, dont les composantes représentent le poids du terme d'indexation considéré dans le document (la requête). Formellement, si on a un espace T de termes d'indexation de dimension N,

$$T = \{t_1, t_2, \ldots . t_j, \ldots . t_n\}.$$

Un document d_i est représenté par un vecteur

$$d_i\,(w_{i1},\, w_{i2},\ldots\ldots,w_{ij},\ldots..w_{in}).$$

Une requête q par un vecteur

$$q\,(w_{q1},\, w_{q2},\ldots..\, w_{qj},\ldots.w_{qn})$$

Où w_{ij} $(\text{resp. } w_{qj})$ représente le poids du terme t_j dans le document d_i (respectivement dans la requête q).

Le modèle vectoriel offre des moyens pour la prise en compte du poids de terme dans le document.

Dans la littérature, plusieurs schémas de pondération ont été proposés. La majorité de ces schémas prennent en compte la pondération locale et la pondération globale **[Zhai, 2010]**.

La pondération locale permet de mesurer l'importance du terme dans le document. Elle prend en compte les informations locales du terme qui ne dépendent que du document. Elle correspond en général à une fonction de la fréquence d'occurrence du terme dans le document (notée *tf* pour *t*erm *f*requency), exprimée ainsi :

$$tf = \log\,(f(t,d))$$

$$tf = \log(f(t,d)+1)$$

$$tf = f(t,d)/\max(\big(f(t,d)\big)$$

Où $f\,(t,\, d)$ est la fréquence du terme t dans le document d.

Quant à la pondération globale, elle prend en compte les informations concernant le terme dans la collection. Un poids plus important doit être assigné aux termes qui apparaissent moins fréquemment dans la collection. Car les termes qui apparaissent dans de nombreux documents de la collection ne permettent pas de distinguer les documents pertinents des documents non pertinents (i.e. peu utile pour la discrimination).

Un facteur de pondération globale est alors introduit. Ce facteur nommé *idf* (*i*nverted *d*ocument *f*requency), dépend d'une manière inverse de la fréquence en document du terme et exprimé comme suit :

$$idf = \log\frac{|N|}{n}$$

$$idf = \log\frac{|N - n|}{n}$$

Où n est la fréquence en document du terme considéré, et N est le nombre total de documents dans la collection.

Les fonctions de pondération combinant la pondération locale et globale sont référencées sous le nom de la mesure :

$$tf * idf = \log(1 + tf) * \log\frac{|N|}{n}$$

Cette mesure donne une bonne approximation de l'importance du terme dans les collections de documents de taille homogène. Cependant, un facteur important est ignoré, la taille du document.

En effet, la mesure (*tf* x *idf*) ainsi définie favorise les documents longs, car ils ont tendance à répéter le même terme, ce qui accroit leur fréquence, par conséquent augmente la similarité de ces documents vis-à-vis de la requête.

Pour remédier à ce problème, des travaux ont proposé d'intégrer la taille du document dans les formules de pondération, comme facteur de normalisation **[Singhal, 1996].**

I.2.3.3 Le modèle probabiliste

I.2.3.3.1 Le modèle probabiliste de base

Le modèle probabiliste est fondé sur la théorie des probabilités **[Robertson, 1977].** Il trie les documents selon leur probabilité de pertinence vis-à-vis d'une requête. La fonction de classement (tri) de ce modèle est exprimée ainsi :

$$RSV\,(q\ ,\ d) = p\ (\ per \mid q\ ,\ d_i) \,/\, p\ (\ Nper \mid q\ ,\ d_i\ )$$

L'idée de base de cette fonction est de sélectionner les documents ayant à la fois une forte probabilité d'être pertinents et une faible probabilité d'être non pertinents à la requête.

Où p (per | q , d_i) et p (Nper | q , d_i) : la probabilité qu'un document $\mathbf{d_i}$ soit pertinent (per) vis à- vis de la requête **q** (respectivement non pertinent (Nper)).

$$P(per|q, d_i) = (P(per|q) \cdot p(d_i | per, q)) / p(d_i)$$

$$P(NPer|q , d_i) = (p(Nper|q) \cdot (d_i | Nper, q)) / p(d_i)$$

Où : p ($d_{i)}$ est la probabilité de choisir le document d_i, on considère qu'elle est constante ; p (d_i | per, q) indique la probabilité que d_i fait partie des documents pertinents pour la requête q ; P (d_i | Nper, q) indique la probabilité que di fait partie des documents non pertinents pour la requête q ; p (per | q) et p (Nper | q) indiquent respectivement la probabilité de pertinence et de non pertinence d'un document quelconque (avec (per | q) + p (N per | q) = 1) qui sont fixes.

Après remplacement dans la fonction de tri, on aura la formule suivante :

$$RSV(q, d) = p(di | per, q) / p(di | Nper, q)$$

I.2.3.3.2 Le modèle de langue

Les modèles statistiques de langue sont exploités avec beaucoup de succès dans divers domaines : la reconnaissance de la parole **[Jelinek, 1998]**, la traduction automatique **[Brown, 1993]**, la recherche d'information **[Croft, 1998]**, etc. L'utilisation des modèles de langue en RI remonte à 1998 **[Croft, 1998]**. Le principe de ce modèle consiste à construire un modèle de langue pour chaque document, soit M_d, puis de calculer la probabilité qu'une requête q puisse être générée par le modèle de langue du document, soit p (q | M_d). Le modèle de langue utilisé est souvent le modèle uni-gramme, la probabilité p (q | M_d) est alors exprimée ainsi :

$$p(q | M_d) = \prod_{t \in q} p(t | M_d)$$

p (t | M_d) peut être estimée en se basant sur l'estimation maximale de vraisemblance.

Elle est donnée par :

$$P(t | M_d) = tf(t, d) / |d|$$

Où $tf(t, d)$ est la fréquence du terme ti dans le document d et |d| est la taille de document.

Pour remédier au problème posé par les mots de la requête absents dans le document, qui ont pour effet d'avoir la probabilité p (t | M_d) nulle ; des techniques de lissage (smoothing) sont utilisées, dont le lissage de Laplace (ajouter-un), le lissage de Good-Turing, le lissage Backoff, le lissage par interpolation **[Zhai, 2001]**, etc. Leur principe consiste à assigner des probabilités non nulles aux termes, qui n'apparaissent pas dans les documents **[Boughanem, 2004]**.

I.2.4 La reformulation de la requête

La reformulation de la requête est un processus permettant la construction d'une nouvelle requête, pour mieux représenter les besoins en information de l'utilisateur. Elle est souvent opérée par ajout et/ou réévaluation des poids des termes de la requête initiale.

Les techniques de reformulation de la requête peuvent être classées en tenant en compte de plusieurs paramètres **[Carpineto, 2012]** :

- Les sources de données utilisées pour l'expansion de requête.

- La méthode de sélection des termes d'expansion : la relation de cooccurrence, les mesures d'information, les techniques de classification, etc.

- La sélection des termes d'expansion en considérant chaque terme de la requête individuellement, ou la requête dans son ensemble.

- La représentation de la requête (document) comme un ensemble de mots simples (sac de mots) ou une représentation prenant en compte les relations de proximité entre termes.

La reformulation de la requête peut être réalisée par l'utilisateur, dans ce cas elle est dite manuelle, ou par le système (dite automatique) comme elle peut être réalisée conjointement par l'utilisateur et le système, dans ce cas elle est dite semi-automatique.

I.2.4.1 Reformulation par réinjection de la pertinence

Ces méthodes impliquent que l'utilisateur doit sélectionner les documents qu'il considère pertinents à partir des résultats issus de sa requête initiale. Ce jugement de pertinence de l'utilisateur est ensuite exploité pour reformuler la requête initiale en

modifiant le poids des termes qu'elle contient et/ou en ajoutant de nouveaux termes considérés utiles pour retrouver des documents pertinents.

La technique de réinjection de pertinence a été mise en place à l'origine dans le modèle vectoriel **[Rocchio, 1971]**.

I.2.4.2 La réinjection par pseudo feedback (réinjection aveugle)

Ces méthodes de reformulation nommées aussi, pseudo-réinjection de pertinence (ou blind) sont effectuées de manière automatique. Elles se basent sur l'hypothèse que les documents les mieux classés (les premiers) sont considérés comme pertinents. Le système utilise alors les premiers documents pour reformuler la requête.

Plusieurs travaux **[Harman, 1992]** ont tenté d'évaluer l'impact de la pseudo-réinjection, en variant le nombre de termes à rajouter à la requête. Ils montrent que la performance du système est obtenue lorsque la requête est construite entre 20 et 40 termes.

I.3 Evaluation des SRI

Dès l'apparition des premiers SRI, la pratique d'évaluation des systèmes est apparue; les premières évaluations datent de 1953 **[Lancaster, 1979]**.

L'évaluation des SRI est abordée selon deux angles différents. L'un est dit « paradigme système », qui vise à évaluer les performances du système essentiellement en termes de qualité des documents retournés par le système, c'est-à-dire leur pertinence vis-à-vis des besoins en information des utilisateurs. L'autre est dit « paradigme usager », qui est centré sur la satisfaction de l'utilisateur, et non sur les performances intrinsèques du système, en modélisant le comportement des utilisateurs en situation de recherche.

Nous présentons ci-dessous seulement l'approche basée « système », la plus utilisée dans le domaine de la RI. Elle se base sur deux éléments essentiels à savoir : des collections de test et des mesures d'évaluation.

I.3.1 Les collections de test

Une collection (ou corpus) de test constitue le moyen d'évaluation des SRI. Elle est généralement composée d'un ensemble de documents, d'un ensemble de requêtes et des jugements de pertinence associés à ces requêtes. L'évaluation d'un SRI consiste à comparer les résultats retournés par ce dernier par rapport aux jugements de

pertinence. Des mesures d'évaluation, décrites dans la section suivante, sont utilisées pour effectuer cette comparaison **[Voorhees, 2005]**.

I.3.2 Mesures d'évaluation de SRI

Le principal objectif d'un système de recherche d'information est de restituer à l'utilisateur tous les documents pertinents et de rejeter tous les documents non pertinents. Cet objectif est évalué à l'aide de différentes mesures d'évaluation **[Sanderson, 2010]**. On présente ci-dessous les plus utilisées.

- **La précision** : est le rapport du nombre de documents pertinents restitués par le système

(SP) sur le nombre total de documents restitués (R), exprimée ainsi :

$$\text{Précision} = \frac{SP}{R}$$

- **Le rappel** : est le rapport du nombre de documents pertinents restitués (SP) sur le nombre total de documents pertinents (P), exprimé ainsi :

$$\text{Rappel} = \frac{SP}{P}$$

Des mesures complémentaires au rappel et précision ont été définies, il s'agit de bruit et de silence.

- **Le bruit** : la mesure d'évaluation bruit est une notion complémentaire à la précision, elle est définie par $B = 1 - P$ où P est la précision du SRI.

- **Le silence** : la mesure d'évaluation silence est une notion complémentaire au rappel, elle est définie par $S = 1 - R$ où R est le rappel du SRI.

I.4 Conclusion

Dans ce chapitre nous avons passé en revue les principaux concepts de la RI. Nous avons, particulièrement, introduit des notions de base, telles que le besoin en information, la requête, le document et la pertinence. Nous avons aussi décrit les processus de base de la RI, à savoir l'indexation, l'appariement requête-document et la reformulation de la requête. Ensuite, nous avons étudié les différents modèles de la RI. Enfin, l'évaluation des systèmes de recherche d'information est traitée.

Dans le chapitre suivant, nous abordons la recherche d'information personnalisée.

Chapitre II : La recherche d’information personnalisée

II.1. Introduction :

Compte tenu des limitations des SRI traditionnels qui se trouvent de plus en plus confrontés aux exigences des utilisateurs, les approches en RI se sont orientées vers une nouvelles génération des SRI basé sur l'accès personnalisé à l'information, leurs objectifs est de mieux répondre aux besoins en information de l'utilisateur, en exploitant le contexte de l'utilisateur ainsi que des connaissances liées à la requête.

La dimension de l'utilisateur est décrite par son profil qui représente ses centres d'intérêts, ses connaissances, et ses buts de la recherche, il est l'élément le plus important qui permet de fournir l'information pertinente qui satisfait intégralement l'utilisateur.

Ce chapitre est constitue de trois grandes parties :

La première partie sur les différents concepts liés à la recherche d'information personnalisé à savoir : la personnalisation, recherche d'information, système de recherche d'information et le profil utilisateur.

La deuxième partie regroupe des techniques d'acquisition des données utilisateur ainsi que des approches de modélisation et de construction du profil utilisateur.

Enfin, la troisième partie concerne l'évolution du profil au cours du temps.

II.2.Les systèmes de recherche d'information personnalisée

II.2.1. Définition

1. **La personnalisation :** est un processus qui change la fonctionnalité, l'interface, la teneur en information, on l'aspect d'un système pour augmenter sa pertinence personnelle en fonction des caractéristiques sociodémographiques déclarées de l'utilisateur (sexe, âge, lieu de résidence, …) et/ou de son comportement. La personnalisation facilite et assiste l'utilisateur lors de sa recherche d'information **[Pernon 00]**.
2. **La RI personnalisée :** est une activité faisant intervenir deux entités principales :
   - Les caractéristiques de l'utilisateur appelées « profil de l'utilisateur ».
   - Les caractéristiques des documents appelées « métadonnées des documents ».
3. **Système de recherche d'information personnalisée (SRIP) :** est un système qui intègre totalement l'utilisateur tout au long de processus de recherche. Il répond ainsi de manière personnelle aux besoins en information de chaque utilisateur.

II.3. Notion de profil

Dans le but de personnaliser l'information, le SRIP doit disposer des éléments ayant une incidence concrète sur la recherche en cours. Ces éléments représentent les données contenues dans le profil utilisateur et les métadonnées des documents.

En effet, l'utilisateur et le document ne sont pas assimilés seulement à des descripteurs exprimés à l'aide de mots comme c'est le cas dans les SRI classiques, ils possèdent tous deux des caractéristiques propres.

II.3.1. Le profil utilisateur

Le profil utilisateur est toute structure qui permet de modéliser et de stocker les données caractérisant l'utilisateur au sein d'un système. Grâce à ce profil, le système possède une connaissance sur les centres d'intérêts, les préférences, et les besoins en informations de l'utilisateur ou un groupe d'utilisateur grâce à lesquelles il peut lui proposer des repenses plus adaptés.

Pour modéliser le profil il faut décrire le « quoi » c'est-à-dire ce qui doit être représenté et le « comment » c'est-à-dire comment représenter le « quoi » au sein du profil **[Amato 99], [Amadieu et Al 09]**.

- Le Quoi : déterminer ce que doit représenter le contenu du profil.
- Le Comment : déterminer la structure de contenu du profil et les techniques utiliser pour la construction de ce profil.

II.3.2. Le profil de document

Le document est l'élément principal d'un SRI, est un objet complexe car il est lié aux développements des technologies de communications.

L'idée principale aujourd'hui, est de renforcer la représentation de l'information du document en utilisant l'information sur l'information du document. En effet, il n'existe pas de représentation unique et donc commune de l'information. Cependant, l'une des étapes primordiales dans un processus de RI c'est l'indexation des corpus documentaires.

Mais, cette indexation ne permet pas de déterminer très précisément les informations contenues par le document. Il est donc difficile pour un SRI de vérifier la pertinence réelle de chaque document. Pour pallier aux limites de l'indexation et pour avoir une meilleure

connaissance du corpus documentaire, les SRIP tentent de décrire les documents par des critères externes à leurs contenus. Ainsi, l'idée est de renforcer la représentation de l'information du document en utilisant l'information sur l'information du document : les métas informations (métadonnées) qui sont utilisés pour définir l'ensemble des informations techniques et descriptives ajoutées aux documents pour mieux les qualifier. Pour que ces données soient utilisables par d'autres, elles doivent s'inscrire dans des modèles largement reconnus par les acteurs du Web...

Plusieurs organismes de standardisation ont donc proposé et publié des schémas de métadonnées susceptibles d'être utilisés par le plus grand nombre.

Le schéma de métadonnées le plus utilisé est proposé par l'organisation Dublin Core Metadata Initiative (DCMI) ; on l'appelle le plus souvent le Dublin Core.

Il standardise l'utilisation d'une quinzaine de champs descriptifs (Titre, Créateur, Sujet et mots clés, ….). Cette stratégie a deux caractéristiques : elle porte sur des critères (la forme, le support, le style, ...) autres que le contenu du document, elle est très fortement individualisée et permet une personnalisation de la recherche

En effet, cette propriété nous permettra de sélectionner un corpus "personnalisé" suivant les caractéristiques de l'utilisateur, corpus sur lequel portera la question.

Les documents sont donc découpés suivant la structure logique du document. Le SRIP utilise ces différentes structures pour décrire les documents en unités documentaires. Chacune des unités est alors accessible par des index bien sûr, mais aussi par ses propriétés. Le découpage est basé sur la fonction remplie par ces parties du document et non sur leur contenu.

L'objectif principal des métadonnées est de décrire, d'identifier et de définir une ressource et de faciliter la recherche d'information **[Pernon 00]**.

II.4. Pertinence contextuelle

La pertinence est incontestablement la question fondamentale posée lors de l'accès à l'information. La pertinence est une notion subjective, non généralisable à tout type d'information et valide pour tout type d'utilisateurs. Elle dépend notamment du centre d'intérêt ou domaine d'application, du moment, du lieu et du support que l'utilisateur a choisi pour accéder à l'information et du système qui délivre cette information.

Cette notion subjective, dépendant essentiellement du point de vue de l'utilisateur, dans le cadre de l'accès personnalisé à l'information. Dans ce contexte, la pertinence est spécifiée comme étant un concept multidimensionnel **[Borlund 03]**, dont on distingue principalement quatre types :

- **pertinence algorithmique** : la pertinence est traduite par une mesure algorithmique dépendant des caractéristiques des requêtes d'une part et des documents d'autre part. C'est le seul type de pertinence qui est indépendant du contexte.
- **pertinence thématique** : la pertinence traduit le degré d'adéquation de l'information à couvrir, en partie, le thème évoqué par le sujet de la requête. C'est le type de pertinence adressé par les assesseurs de la campagne d'évaluation TREC.
- **pertinence cognitive** : c'est la pertinence liée au thème de la requête « Pondérée » par la perception ou les connaissances de l'utilisateur sur ce même thème.
- **pertinence situationnelle** : c'est la pertinence liée à la tâche de recherche. Ce type de pertinence traduit essentiellement l'utilité de l'information relativement au but de recherche de l'utilisateur.

La RI personnalisée explore essentiellement la pertinence cognitive et la pertinence situationnelle.

II.5. Architecture fonctionnelle d'un SRIP [Mariam 09]

Un système de recherche d'information personnalisé (SRIP) est un système qui intègre l'utilisateur, en tant que structure informationnelle, tout au long de la chaîne d'accès à l'information. Le SRIP ne se limite pas seulement à modéliser les caractéristiques des

utilisateurs en des profils, il doit être capable de déduire à partir de ces profils, l'intention de l'utilisateur lorsqu'il effectue sa recherche.

En d'autres termes son contexte de recherche, et de détecter l'évolution des profils de manière dynamique. Le système doit donc inclure :

- Des techniques et algorithmes pour capturer et modéliser le but, les préférences et les centres d'intérêts de l'utilisateur ou un groupe d'utilisateurs. Un modèle de profil utilisateur est alors décrit et instancié.
- Une procédure de mise à jour du profil qui traduit son évolution dans le temps.
- Des mécanismes et algorithmes pour intégrer le profil de l'utilisateur dans le processus d'accès et retourner l'information pertinente en fonction de ce profil.

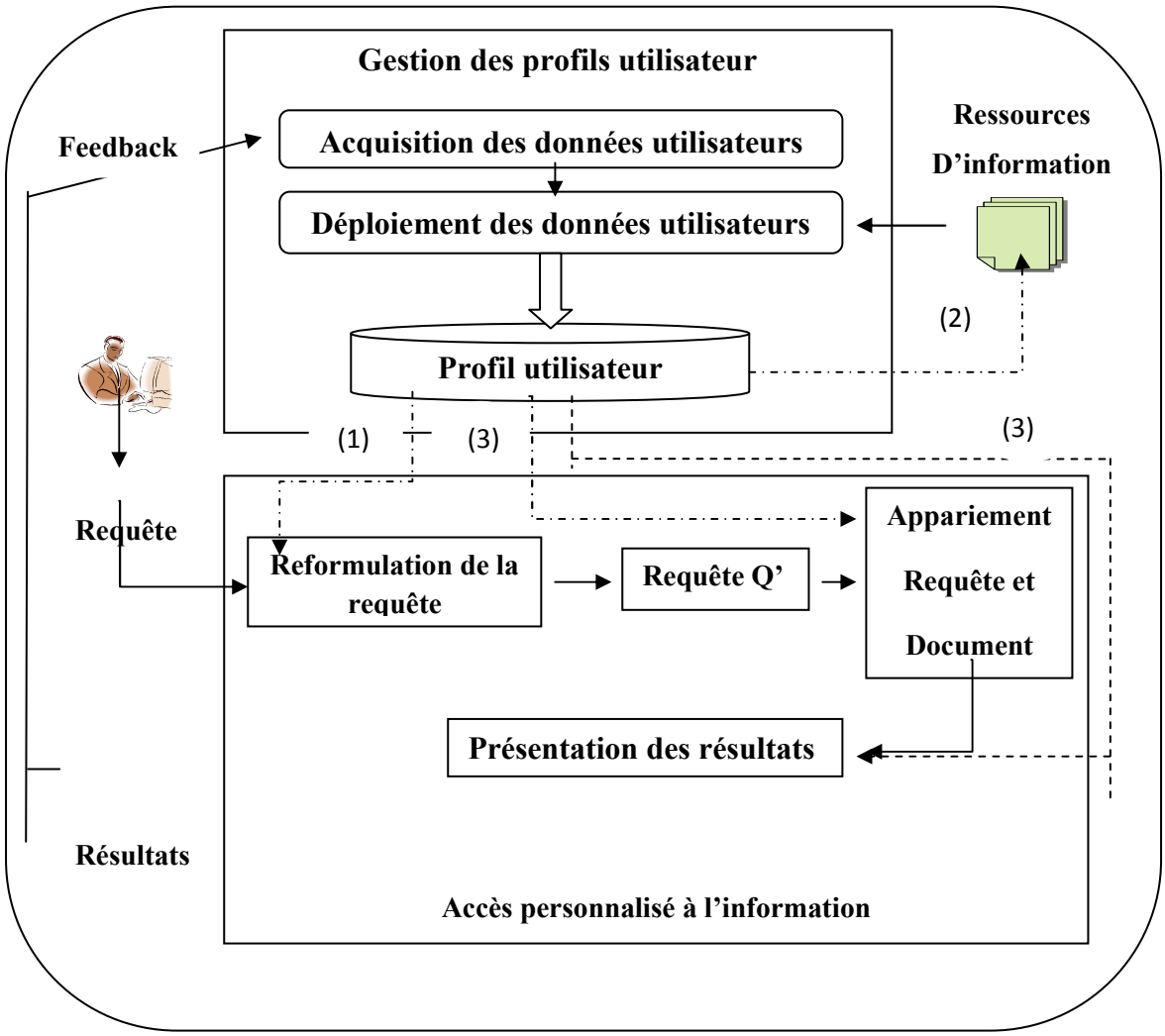

Figure II.1 : Architecture fonctionnelle d'un SRIP

Cette architecture centrée autour de l'utilisateur met en évidence :

1. Un gestionnaire de profil pour représenter, construire et faire évoluer les profils des utilisateurs.

2. Les étapes du cycle de vie de la requête où l'on intègre le profil utilisateur dans :

1) **La phase de reformulation de la requête** afin de mieux cibler le contexte de la recherche de l'utilisateur,
2) **La phase de réduction de l'espace de recherche** pour restreindre l'espace de recherche aux documents qui ciblent les besoins de l'utilisateur,
3) **La phase d'appariement** pour calculer la pertinence des documents en fonction des caractéristiques spécifiques de l'utilisateur,
4) **La phase de présentation des résultats** pour restituer les informations selon le contexte et les préférences de l'utilisateur.

II.6.Mise en œuvre d'un SRIP

II.6.1. Modélisation du l'utilisateur

La modélisation consiste à décrire les caractéristiques informationnelles des utilisateurs à travers un modèle de profil.

Pour modéliser l'utilisateur il faut définir en premier la structure de son profil qui permet non seulement, de stocker les informations le concernant mais aussi de les exploiter d'une manière optimale. En second, il faut déterminer les techniques de construction et de mise à jour de ce profil. Le type d'approche adoptée par le SRIP détermine fortement l'efficacité du système.

II.6.2. Représentation du profil utilisateur

Comme le contenu du profil dépend fortement de l'application qui l'exploite. On distingue trois principales approches de représentation : vectorielle, hiérarchique, et multidimensionnelle **[Mariam 09]**.

1. **Représentation vectorielle**

Ce type de représentation s'appuie généralement sur le modèle vectoriel de Salton **[Salton 71]**, il consiste à représenter le profil de l'utilisateur par des paquets de termes pondérés. Ces paquets de termes, traduisant les centres d'intérêts de l'utilisateur, peuvent

être regroupés différemment selon l'approche suivie pour considérer le profil de l'utilisateur.

On distingue dans la littérature trois grandes approches de représentation du profil utilisateur basées sur ce modèle :

- Par une liste de mots clés, où chaque mot correspond à un centre d'intérêt spécifique
- Par un vecteur de termes pondérés pour chaque centre d'intérêt.
- Par un ensemble de vecteurs de termes pondérés (ou non) indépendants, pour prendre en compte des centres d'intérêt multiples où chaque vecteur correspond à un domaine d'intérêt.

La représentation vectorielle fut parmi les premiers modèles de profils utilisateur exploités en RI.

Cette représentation apporte l'avantage de la simplicité de mise en œuvre. Néanmoins, même si ces systèmes prennent en considération des centres d'intérêts multiples en utilisant plusieurs vecteurs, cette représentation manque de structuration. Cette représentation ne facilite ni l'interprétation ni la prise en compte des différents niveaux de généralités caractérisant l'utilisateur.

Il faut modéliser le profil utilisateur de façon à prendre en considération l'ensemble des paramètres représentant l'utilisateur pour résoudre le problème de l'ordonnancement des préférences et des centres d'intérêts de l'utilisateur.

2. Représentation hiérarchique

La représentation du profil met en évidence, dans ce cas, les relations sémantiques entre informations le contenant. Dans cette approche, la modélisation de l'utilisateur est fondée sur l'élaboration d'une ontologie personnelle. L'ensemble des caractéristiques de l'utilisateur est organisé dans une structure hiérarchique de concepts (catégories) où chaque catégorie représente la connaissance d'un domaine d'intérêt de l'utilisateur.

Le premier à avoir utilisé une telle structure fut Pretschner **[Pretschner 99]** dans le système OBIWAN. Il a proposé un modèle innovant pour la construction du profil utilisateur. Il s'appuie sur l'ontologie publique de Magellam qui est composée

d'approximativement 4.400 nœuds de concepts. Semblable à ce travail, on peut citer le système SmartPush **[Kurki 99]**.

Bien que la représentation de ce profil d'utilisateur soit innovatrice, ces travaux ne se servent pas des caractéristiques de la structure hiérarchique (par exemple pour dédoubler ou fusionner des nœuds dans le profil d'utilisateur) pour capturer la dynamique des changements. De plus, la sémantique générale de cette hiérarchie n'est pas formellement indiquée; dans la plupart des cas, ils correspondent à une relation de généralisation/spécialisation.

3. Représentation multidimensionnelle

La représentation multidimensionnelle du profil s'inscrit dans une réflexion globale sur la personnalisation de l'information. En effet, le profil est structuré selon un ensemble de dimension représentée selon divers formalisme.

Différentes propositions on été abordés pour cet aspect, on a le standards P3P pour la sécurisation des profils ont défini des classes distinguant les attributs démographiques des utilisateurs (identité, données personnelles), les attributs professionnels (employeur, adresse, type) et les attributs de comportement (trace de navigation). Une autre proposition faite par Amato **[Amato 99]** consiste à représenter le contenu du profil utilisateur par un modèle structuré de dimensions (ou catégories) prédéfinis. C'est la première approche où les informations sont structurées et qui offre un modèle général.

Le modèle de profil contient cinq catégories :

1- catégorie de données personnelles,

2- catégorie de données collectées,

3- catégorie de données de livraison,

4- catégorie de données de comportement,

5- catégorie de données de sécurité.

La première catégorie Données personnelles contient toutes les informations concernant l'identité de l'utilisateur.

La deuxième « Données collectées » contient les informations nécessaires pour décrire les préférences et restrictions sur les documents. Elle est divisée en trois sous

catégories : contenu (des informations sur le sujet du document, la langue, etc.) structure, (format, type, date de publication, dimensions, etc.), source (provenance, auteurs, éditeurs, etc.).

Dans la catégorie « Données de livraison », on trouve les informations sur la manière de transmettre des résultats à l'utilisateur. Ces informations sont regroupées selon deux sous catégories : moyen (mode de livraison par exemple email, fax téléphone, etc.) et moment (contient des informations temporelles sur le moment de livraison comme lors d'un changement, vers midi, entre 9h et 9h15, etc.).

Dans la catégorie « Données de comportement » se trouvent des enregistrements sur les interactions de l'utilisateur avec le système (URL des pages visitées, documents lus et pertinence, etc.).

Enfin dans, la catégorie Données de sécurité, des informations sont données sur les conditions d'accès aux données du profil.

Amato et kostadinov **[Amato 99] [Kostadinov 03]**, ils ont proposé un ensemble de dimensions ouvertes, capables d'accueillir la plupart des informations caractérisant un profil.

Il distingue principalement huit dimensions :

1. les données personnelles.
2. le centre d'intérêt.
3. l'ontologie du domaine.
4. la qualité attendue des résultats délivrés.
5. la customisation.
6. la sécurité et la confidentialité.
7. le retour de préférences (feedback).
8. les informations diverses.

Ces classes de données sont brièvement décrites dans ce qui suit :

- Les données personnelles

Les données personnelles sont la partie statique du profil. Elles comprennent l'identité civile de l'utilisateur (nom, prénom, numéro de sécurité sociale, etc.) ainsi que des données démographiques (âge, genre, adresse, situation familiale, nombre d'enfants, etc.)

- Le centre d'intérêt

Le centre d'intérêt exprime le domaine d'expertise de l'utilisateur. Il peut être défini par un ensemble de mots clés ou un ensemble d'expressions logiques (requêtes).

- L'ontologie du domaine

L'ontologie du domaine complète la définition du centre d'intérêts en explicitant la sémantique de certains termes ou de certains opérateurs employés par l'utilisateur dans son profil ou dans ses requêtes.

- La qualité attendue

La qualité est un des facteurs clés de la personnalisation ; elle permet d'exprimer des préférences extrinsèques comme l'origine de l'information, sa précision, sa fraîcheur, sa durée de validité, le temps nécessaire pour la produire ou la crédibilité de sa source. Les attributs de cette dimension expriment la qualité attendue ou espérée par l'utilisateur.

- La customisation

La customisation concerne d'abord tout ce qui est lié aux modalités de présentation des résultats en fonction de la plateforme, de la nature et du volume des informations délivrées, des préférences esthétiques ou visuelles de l'utilisateur.

- La sécurité

La sécurité est une dimension fondamentale du profil. Elle peut concerner les données que l'on interroge ou modifie les informations que l'on calcule, les requêtes utilisateurs elles-mêmes ou les autres dimensions du profil. La sécurité du processus exprime la volonté de l'utilisateur de cacher un traitement qu'il effectue.

- Le retour de préférences

On désigne par ces termes ce qu'on appelle communément le 'feedback' de l'utilisateur. Cette dimension regroupe l'ensemble des informations collectées sur l'utilisateur.

- Les informations diverses

Certaines applications demandent des informations spécifiques ne pouvant être incluses dans aucune des dimensions précédentes comme par exemple la bande passante attribuée au gestionnaire du profil. Pour cette raison l'utilisateur a la possibilité de rajouter ce type de préférences dans la partie divers du profil et de décrire leurs utilisations.

II.6.3. Construction du profil

La constitution d'un profil d'utilisateur ou d'une communauté d'utilisateurs peut se faire de différentes façons selon la nature des informations constituant le profil mais aussi selon la nature des applications.

Elle s'effectue en deux étapes principales :

(1) l'acquisition et la collecte des données utilisateur.

(2) puis la construction proprement dite du profil.

II.6.3.1. L'acquisition et collecte des données du profil

La première phase consiste à collecter les informations pertinentes pour instancier le profil de l'utilisateur. Ce processus peut collecter ces informations soit directement à partir de la machine de l'utilisateur (côté client) ou à partir de l'application (côté serveur). Ce processus d'acquisition peut être explicite et/ou implicite **[W.Zemirli 08]** :

a. L'acquisition explicite

Cette technique constitue une approche simple pour obtenir des informations sur l'utilisateur. On interroge directement l'utilisateur ou on lui demande par exemple de remplir des formulaires pour collecter ses préférences sur les dimensions, les membres ainsi que des informations décrivant son environnement à savoir la taille de son écran, la vitesse de son processeur et la taille de sa mémoire, etc.

En effet, l'utilisateur émet directement son jugement d'intérêt en donnant une valeur de pertinence sur une échelle graduée allant du moins intéressant au plus intéressant.

b. L'acquisition implicite

L'acquisition implicite ou « feedback implicite » consiste à collecter les informations décrivant l'utilisateur, en observant les dimensions et les membres

fréquemment sollicités et en scrutant les caractéristiques de l'environnement à partir duquel il intervient (les capacités et les limites du dispositif utilisé lors de ses interactions). Et ce, en se basant sur l'historique de ses interactions avec le système.

Le principal avantage de cette approche est qu'elle ne nécessite aucune implication directe de l'utilisateur, ni de temps passé à émettre des jugements, ni un effort d'attention particulier lors de son interrogation. On peut noter aussi que la sécurité et la confidentialité des informations sont préservées.

L'inconvénient se trouve dans la complexité des algorithmes utilisés qui nécessitent beaucoup de temps.

II.6.3.2. La construction du profil

La construction du profil requière plusieurs techniques de conception : les techniques de construction d'un profil initial, la source de feedback qui représente les intérêts de l'utilisateur et les techniques d'apprentissage des profils.

II.6.3.2.1. La génération du profil initial

Intuitivement la manière la plus simple d'initialiser un profil est la saisie manuelle des paramètres par l'utilisateur. Il est considéré que l'utilisateur connaît mieux ses exigences et de ce fait, il peut saisir les paramètres dont a besoin le système. On peut exposer quelques méthodes d'obtention des informations sur les utilisateurs **[I.Boussaid 05]** :

II.6.3.2.1.1. L'observation directe

Il s'agit de la méthode la plus précise. Elle permet d'identifier des classes d'utilisateurs ainsi que les tâches de ces derniers. Malheureusement, il s'agit d'une méthode très coûteuse qui nécessite des personnes qualifiées derrière chacun des individus observés.

II.6.3.2.1.2. Les interviews

Cette technique permet d'obtenir un autre type d'information, l'expérience, les opinions, les motivations comportementales mais surtout les avis sur les outils existants. Ils sont plus courts et moins coûteux que la technique d'observation, néanmoins, ils nécessitent aussi du personnel qualifié.

II.6.3.2.1.3. Les questionnaires

Les questionnaires permettent d'obtenir, à moindre coût, un maximum de données. Les résultats obtenus permettent des études statistiques et des généralisations plus fortes que les interviews. Les questionnaires peuvent être collectés par des personnes non expérimentées. Ils permettent d'avoir à la fois un aperçu de la situation et des points d'information plus précis.

Cependant, la tendance est à minimiser les actions de l'utilisateur parce que le processus de saisie manuelle peut être long et bien souvent l'utilisateur a des idées floues sur ses demandes et par conséquent a du mal à exprimer clairement ses intentions, et il est très possible que cet utilisateur fausse lui-même les données le concernant. C'est en cela que la collection informatisée d'informations sur l'utilisateur est également délicate.

Afin de résoudre ces problèmes, ils existent plusieurs méthodes de capture des paramètres du profil de façon semi-automatique (à travers des stéréotypes et des données d'apprentissage) ou automatique (méthodes liées à l'interprétation des activités de l'utilisateur).

II.6.3.2.2. Techniques d'apprentissage des profils

Il est possible d'obtenir de l'information par l'intermédiaire d'un outil d'apprentissage. En effet, il est intéressant, et très utile d'y ajouter un algorithme d'apprentissage pour obtenir des informations essentiellement comportementales sur l'utilisateur.

II.6.3.2.2.1 Analyse statistique de termes

Ce processus peut être vu comme un prétraitement des données et leur mise en forme normale pour pouvoir les manipuler ensuite. L'idée principale consiste à analyser le contenu d'un document et d'extraire des mots clés significatifs qui décrivent son contenu. Ces mots clés sont stockés pour être utilisés ensuite afin de comparer des éléments entre eux ou avec les préférences de l'utilisateur lorsqu'elles sont exprimées sous la forme de mots clés. Il existe différentes structures de stockage et de représentation des mots clés en fonction du contexte dans lequel le profil est utilisé

Il existe différentes structures de stockage et de représentation des mots clés en fonction du contexte dans lequel le profil est utilisé **[Soltysiak et Crabtree, 1998]**. En plus dans certains cas, on peut rajouter un poids qui exprime l'importance de chaque terme et qui est souvent associé à la fréquence d'apparition du terme.

Un terme est ici une suite de caractères alphanumériques délimités par des espaces ou des ponctuations. Les majuscules sont souvent converties en minuscules. Après avoir extrait ces termes du corpus, chaque document est alors représenté par un vecteur où chaque terme est pondéré selon une fonction de poids.

II.6.3.2.2.2. Les techniques de classification

Le but de la classification est d'attribuer un élément donné à un groupe existant. Les groupes sont connus à l'avance et sont donnés en paramètre à l'algorithme qui ensuite attribue les objets à un groupe selon certains critères.

Les algorithmes de classification les plus utilisés dans la littérature sont le raisonnement par cas, les classificateurs bayésiens les réseaux de neurones et les règles d'associations **[K.Al Makssoud 08]**.

II.6.3.2.2.3. Les méthodes de clustering

A la différence de la classification, dans les techniques de clustering les groupes d'objets ne sont pas connus à l'avance et c'est l'algorithme qui se charge de la répartition des éléments en essayant de minimiser la similarité entre les éléments de deux clusters différents et de maximiser la similarité entre les éléments du même cluster **[K.Al Makssoud 08]**.

Les deux principales techniques de clustering sont :

La première PACT (Profile Aggregation based on Clustering Transactions) consiste à regrouper les transactions similaires d'un utilisateur. Chaque transaction est un vecteur multidimensionnel de pageviews (vues sur les pages Web), et le regroupement est fait à la base de la distance ou de la similarité entre les vecteurs.

Le second ARHP (Association Rule Hypergraph Partitioning) prend en compte les pageviews apparaissant souvent ensemble. ARHP peut être utilisé lorsqu'on veut produire un petit ensemble de recommandations spécifiques.

Les deux techniques (PACT et ARHP) peuvent être utilisées pour une personnalisation basée sur les données de navigation des utilisateurs.

II.6.3.2.3 Retour de pertinence (Relevance feedback)

Par un processus de retour de pertinence (« Relevance feedback » en anglais), l'utilisateur n'indique pas seulement l'information pertinente mais aussi l'information non-pertinente.

Le système utilise ces informations pour ajuster la description du profil utilisateur qui reflètera les nouvelles préférences. Cette procédure provoque une évolution constante du profil utilisateur. Ce profil se stabilise après plusieurs recherches, une fois que le profil est défini au plus proche des préférences utilisateur. Bien qu'ils existent déjà plusieurs techniques permettant de capturer les préférences d'un utilisateur, leur utilisation n'est pas encore bien comprise pour permettre la construction et la mise à jour de son profil de façon automatique et complètement transparente pour lui. Actuellement nous pouvons espérer que ceci se fasse avec une implication minimale de l'utilisateur. Le principe est que l'utilisateur doit avoir le contrôle de son profil à tout moment afin de pouvoir invalider les mises à jour incorrectes du profil. Après une telle invalidation, le gestionnaire doit prendre en compte et modifier la manière (le processus) de gestion et de mise à jour du profil.

II.7. Intégration du profil utilisateur dans le processus de recherche d'information

II.7.1. Intégration du profil utilisateur dans la phase de présélection de l'espace de recherche

A partir des informations continues dans le profil le SRIP va cibler un ensemble de documents. Ces informations sont représentées par une liste de termes, cette liste va être comparée aux descripteurs des documents de la collection. Les documents ayant un degré de similarité élevé seront sélectionnés et formeront le sous espace de recherche sur lequel portera la requête.

II.7.2. Intégration du profil dans la phase d'évaluation de requête

La requête est l'expression d'un besoin circonstancié que l'utilisateur souhaite voir satisfait en tenant compte de son profil. Elle représente le besoin à court terme de l'utilisateur, alors que le profil représente son besoin à long terme.

La requête initiale de l'utilisateur contient des valeurs de paramètres du profil utilisateur fournies directement par l'utilisateur. Ces paramètres représentent les classes d'intérêt de haut niveau dans la structure hiérarchique de profil utilisateur En choisissant une classe d'intérêts l'utilisateur exprime son but de recherche. A partir des choix le système va extraire les documents correspondants et effectuer la reformulation de la requête. En effet, le SRIP va considérer ces documents comme jugés pertinents afin de reformuler la requête initiale.

II.7.3.Intégration du profil dans la phase de présentation du résultat

Dans cette phase l'interaction entre le profil et le module de représentation du résultat se fait à travers les catégories « Customisation » et « Catégories des données de l'environnent ». En fonction des informations fournies par ces catégories le SRIP va déployer des mécanismes adaptation afin de bien restituer ces résultats selon le contexte et les préférences de l'utilisateur.

II.7.4. Intégration du profil dans la phase de réduction de l'espace de recherche

La phase de réduction de l'espace de recherche consiste à cibler dans l'ensemble du corpus géré par le SRIP, le sous-ensemble de documents susceptibles d'être pertinents en fonction du profil utilisateur. Le système présélectionne les documents ayant un degré de similarité élevé pour former un corpus personnalisé de documents qui sera exploité lors de la phase d'évaluation de la requête.

II.8. Évolution du profil utilisateur

L'évolution du profil utilisateur est un processus complémentaire à la construction d'un profil utilisateur et désigne leur adaptation à la variation des centres d'intérêts des utilisateurs au cours du temps.

L'évolution du profil utilisateur consiste principalement à capturer les changements des centres d'intérêts de l'utilisateur dans une première phase et propager ces changements au niveau de la représentation du profil.

Les techniques de collecte des informations utilisées dans le processus de l'évolution du profil utilisateur sont relativement dépendantes de la portée temporelle du profil, on distingue alors le profil à court terme et le profil à court terme.

L'évolution du profil à court terme [Gowan 03] représente les centres d'intérêts liés aux activités de recherches courantes de l'utilisateur. Il sert à mieux cibler la recherche puisqu'il contient des informations considérées spécifiques et pertinentes au besoin en information de l'utilisateur. Le but fondamental de l'évolution du profil à court terme est d'améliorer la précision de recherche en utilisant le profil le plus utile et approprié à la requête.

L'évolution du profil à long terme représente les centres d'intérêts persistants de l'utilisateur et issus de son historique de recherche tout entier. Il consiste à ajouter ou modifier un profil préalablement appris selon des changements éventuels des centres d'intérêts de l'utilisateur au cours des sessions de recherche **[Sieg, Mobasher, Burke, Prabu et Lytinen 04]**.

II.9. Conclusion

Ce chapitre consiste en premier temps à définir les différentes notions de base des systèmes de recherche d'information personnalisée, la notion de profil et de documents qui sont des concepts clés de la **RIP**.

De plus, nous avons abordé le concept de modélisation de l'utilisateur incluant les approches de représentation du profil, les techniques de construction et son intégration dans le processus de recherche d'information la représentation de ce dernier dans un modèle ou par une structure qui permet son exploitation par le SRI.

Le chapitre suivant présente notre approche d'intégration de profil utilisateur dans la phase d'appariement d'un système de cherche d'information.

Chapitre III : Conception De Système de RI Personnalisé

III.1 Introduction

Le présent chapitre portera sur la modélisation du profil utilisateur et son intégration dans la phase d'appariement du processus de recherche d'information**.** Nous allons commencer par présenter l'architecture du système personnalisé à concevoir avant de procéder à son étude détaillée : soit la description de ses différents modules et les algorithmes ressorties.

III.2 Architecture du Système de recherche d'Information Personnalisé

Tel qu'il a été bien dit dans le chapitre précédent, le processus de personnalisation par **l'intégration du profil utilisateur** touche à l'une ou à toutes les phases du processus de recherche.

Le modèle d'appariement de notre système se base sur un modèle de RI vectoriel où le composant profil utilisateur est représenté par deux unités informationnelles : le centre d'intérêt et les préférences de recherche (exemple : format de document à rechercher qui peut être : PDF, html, doc,…etc.).

Notre système regroupe 4 modules (voir Figure 3.1):

1- Module de Requête

2- Module d'Indexation

3- Module Profil

4- Module Recherche

De plus, le système fait intervenir deux statuts (Simple utilisateur et Administrateur). Nous décrivons ici les tâches attribués à chaque statut.

L'Administrateur, après authentification au système, peut :

- Ajouter et indexer un document ;
- Catégoriser un document ;
- Supprimer un document ;
- Consulter la base de données

A un simple utilisateur sont offertes d'autres tâches. Une interface d'interrogation lui permet de :

- S'authentifier ;

- S'inscrire et spécifier son centre d'intérêt (domaine de recherche, sous domaine) ; préférences de recherche (formats de documents récupérés) ;

- Effectuer une recherche personnalisée ;

- Visualiser les résultats ;

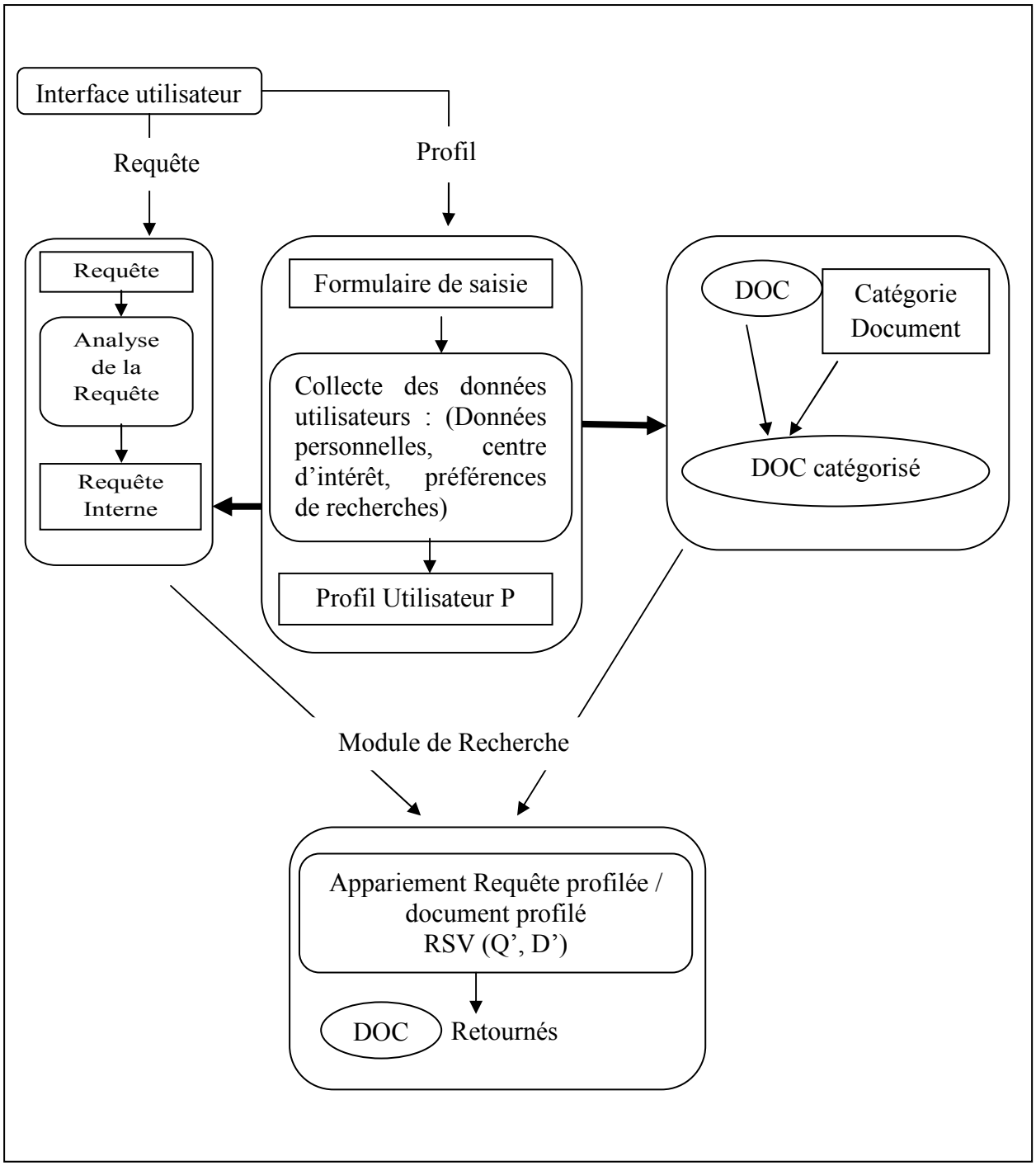

Figure III.1 : Architecture du Système de Recherche d'Information Personnalisé

III.3 Modélisation du profil utilisateur

L'objectif fondamental d'un système de recherche d'information basée sur le profil utilisateur est de retourner, à partir d'une collection de documents, les éléments qui sont pertinents à un besoin en information exprimé par l'utilisateur à travers une requête. La sélection des seuls documents intéressants un utilisateur se fait sur la base des données collecter sur l'utilisateur appelé profils et de la représentation des documents sous formes d'un index. La modélisation de ce profil est la clé de réussite de tout le processus de recherche dont la finalité est de satisfaire l'utilisateur en quête d'information en ne rapportant pour lui que les documents pertinents susceptibles de l'intéresser. Pour construire le profil utilisateur nous avons adopté dans notre travail une représentation multidimensionnelle qui concerne les déférents types d'information de l'utilisateur ainsi qu'une représentation de ces centres d'intérêts.

III.3.1 Module d'indexation :

En fonction du profil utilisateur, le système va regrouper la collection de documents susceptible d'intéresser l'utilisateur. Ceci est rendu possible à l'aide d'une indexation classique des documents, augmentée par une indexation de catégories de documents. Un document donné est catégorisé par un nom de catégorie (qui renseigne sur la thématique traité dans le document). Celle- ci est donc apparié lors du processus de recherche avec le centre d'intérêt de l'utilisateur, d'où la collection de document profilé. (Voir Figure 3.1).

III.3.2 Module Requête :

Pour cibler le besoin de l'utilisateur, le système analyse la requête. Il est clair que si l'utilisateur formule sa requête initiale en tenant compte de ce qu'il est c'est-à-dire de son profil ; le système pourra mieux interpréter son besoin et donc mieux le satisfaire. Partant de cette constatation, notre idée est d'essayer au maximum d'avoir une requête « pointue » dés le début de la recherche. A cet effet, le système demande à l'utilisateur de renseigner son centre d'intérêt (domaine ou thématique de recherche) et quelques préférences sur le format des documents à retourner. Ainsi, la requête ne sera plus constituée d'une simple liste de mots clés que l'utilisateur formule, mais également des valeurs de propriétés du profil. La requête ainsi améliorée par les données du profil est nommée « *requête profilée* ».

III. 3.3. Module Recherche :

En exploitant ce qui a été dit sur les modules précédents, le système effectue alors un appariement entre la nouvelle requête étendue en fonction du profil et la collection documentaire personnalisé.

III.4 Intégration du profil utilisateur

La Figure ci- après (Figure 3.2) explicite l'intégration du profil utilisateur dans le processus de recherche : de l'expression des besoins à la recherche proprement dite, passant par l'indexation et la personnalisation de la collection documentaire. Rappelons que nous notre choix est de construire un SRIP sur un modèle de RI vectoriel où la requête et le document sont représenté par des vecteurs. Soit *Q* une requête, *P* un profil utilisateur et D un document de la collection documentaire.

1) L'analyse de la requête prend en entrée les deux paramètres *(Q, P)* et en sortie la requête profilée *Q'*. La requête profilée n'est qu'un vecteur de termes pondérés, soit :

Q'= (t1, t2, ..tn), n : nombre des termes de la requête.

Dans notre cas, tous les termes de la requête ont le même poids.

2) D'un autre coté, les documents sont indexés par catégorie. Il est possible alors lors d'un processus de recherche de réduire l'espace de recherche en sélectionnant seulement les documents dont la catégorie est en correspondance avec le profil utilisateur (dans notre cas son centre d'intérêt).

A partir de 1) et 2), la pertinence entre la requête profilée et le document profilé est mesurée en se basant sur la mesure :

$$RSV(Q', D') = \sum_{i=1}^{n} (wQj * wij)$$

Où i = indice du document, j = indice du terme, n = longueur du document.

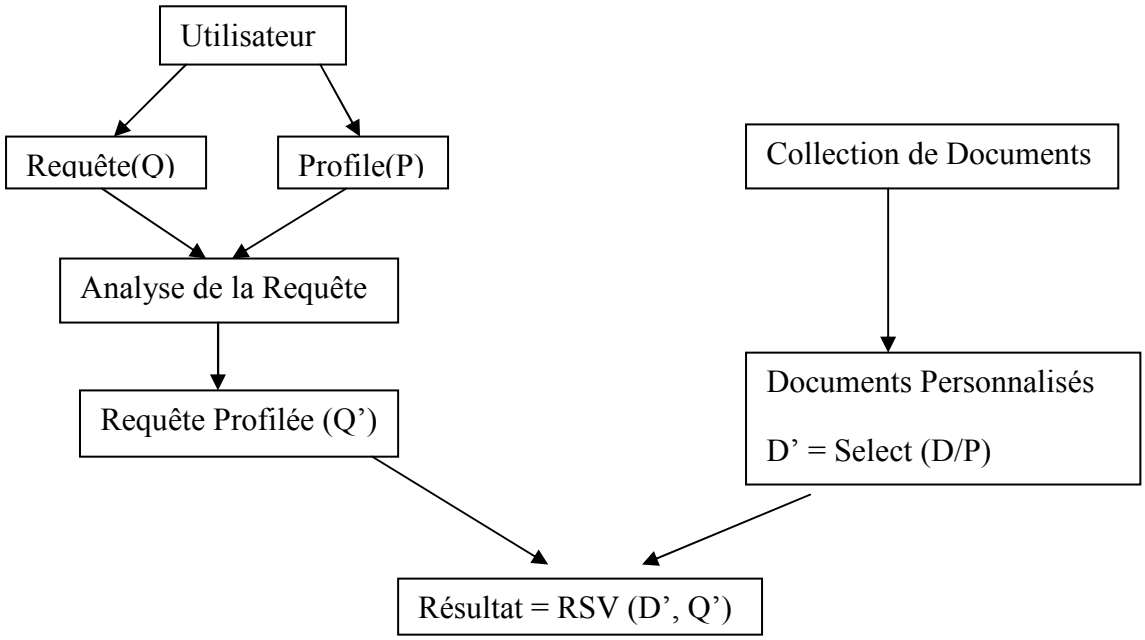

Figure III .2: Intégration du profil utilisateur dans le processus de recherche d'information.

III.5 Architecture de la base de données

Pour stocker les index et issus de la phase d'indexation, et le profil des utilisateurs nous avons construit une base de données contenant les tables suivantes :

administrateur (Login, mot_de_passe) ;

table_utilisateur (ID_User, Nom, Prenom, Login, Mot_p, Domaine_interet, Format_doc) ;

table_preferences (ID_User, Format_Doc);

tab_collection (id_col,nbr_doc,nbr-terme,type,adr_collection);

tab_doc (Id_doc,document,Taille_doc,adr_doc,cat_doc) ;

tab_fich_dir (Id_doc,doc,terme_doc,freq_terme) ;

tab_fich_inver (Id_doc,terme_doc,doc ;freq_terme) ;

tab_freq_doc (terme,nb_doc) ;

III.6 Interprétation algorithmique

III.6.1 Module Requête :

Méthode *Analyse_Request ()*

Entrées : Chaine de caractères : Requête, centre d'intérêt, les préférences en recherche ;

Sorties : termes de la Requête profilée ;

Variables : Listes : list_termes_req, list_termes_domaine, list_termes_format ;

Chaine de caractères : token, terme ;

Début

Extraire les token de la chaine requête ; // les mots constituant la requête.

//Vérifier si le token appartient à la StopList (StopList est la liste des mots vides construite préalablement).

Pour chaque token **faire**

Si (token StopList) **alors**

Ignorer le token ;

Sinon

terme *trimword* (token) ;

Ajouter le terme à la list_termes_req ;

Fin Pour

Refaire la même chose pour analyser le centre d'intérêt et les préférences de recherche et sortir avec les termes de la requête profilée.

Fin

Méthode *trimword ()*

Entrée : token ; *// un mot*

Sortie : terme ;

Initialisation : terme token ;

Début

Si le token commence par **; ou , ou : ou . alors**

terme← Eliminer le premier caractère du token ;

Si le token se termine par **; ou , ou : ou . alors**

terme← Eliminer le dernier caractère du token ;

Si (premier_caractère_token d) **et (**second_caractère_token) **') ou**

(premier_caractère_token l) **et (**second_caractère_token) **') alors**

terme ← Eliminer le premier et le second caractère du token ;

Si (** la longueur du terme > 7) **alors

terme ← Retenir que les 7 premières lettres du terme ;

Fin

III.6.2 Module d'indexation :

Méthode *insert_collection ( )*

Entrées : chemin de l'emplacement collection, catégorie de la collection ;

Variables : Chaines de caractères : adr_collection, type, nbr-terme,nbr_doc ;

Début

Ouvrir la connexion a la base de données ;

setData(nbr_doc,nbr-terme,type,adr_collection);//insère à la tab_collection

Fermer la connexion à la base de données ;

Fin

getData (resp. *setData*) est une méthode prévue pour exécuter des requêtes de sélection de données (resp. de modification de données) de type SQL.

Méthode *index_document ( )*

Entrées : chemin de l'emplacement du document ;

Sorties : les index du document ;

Variables : Listes : list_termes_doc , list_freq_temes, document,Taille_doc,adr_doc,cat_doc ;

Entier : Cmpt ; // compte le nombre de termes

Début

Initialiser un buffer pour contenir le document à lire ;

Tanque document existe **faire**

Debut

Ouvrir la connexion a la base de données ;

setData (document,Taille_doc,adr_doc,cat_doc) ;);//insère à la tab_doc

Fermer la connexion à la base de données ;

Lire (document) ;

Pour chaque ligne du document **répéter**

Extraire les tokens de la ligne ;

Pour chaque token **faire**

Si (token ϵ StopList) **alors**

Ignorer le token ;

Sinon

terme ← *trimword* (token) ;

Si (terme ϵ list_termes_doc) **alors**

i ← list_terme_doc (terme) ;

// recuperer l'indice du terme dans la liste

Récupérer la fréquence du terme à l'indice i ;

Incrémenter la fréquence à l'indice i ;

Sinon

Ajouter le terme à la list_termes_doc ;

i ← list_terme_doc (terme) ;

// recuperer l'indice du terme dans la liste

list_freq_termes (i) ← 1 ;

// la première occurrence du terme, donc la fréquence reçoit 1

cmpt ← cmpt +1 ;

// incrementer le nombre de termes

Fin pour

Fin pour

setData (terme,nb_doc) //Insérer à la table tab_freq_doc ;

setData(Id_doc,doc,terme_doc,freq_terme)//Insérer à la table tab_fich_der ;

setData(Id_doc,terme_doc,doc ;freq_terme) //Insérer à la table tab_fich_inver ;

Fin tanque

Fin

III.6.3 Module Recherche :

Méthode *appariement ( )*

Entrée : les documents indexés, les catégories de documents indexés, la requête profilée ;

Sortie : pertinence des documents ;

Initialisation : RSV(d, Q) ← 0 ;

//Tous les poids des termes de la requête sont pondérés à 1 ;

Début

//récupérer les termes de la requête, les termes du centre d'intérêt.

Lire (requête profilée) ;

// Construire la collection de documents profilé

//récupérer le nb_doc profilé

Sélectionner les ID_doc tel que le terme catégorie est égal au terme résultant de l'analyse du centre d'intérêt.

Lire (termes_requete) ;

Pour chaque terme tj de la requête **faire**

Pour chaque terme tij du document sélectionné **faire**

Si (tj = tij) **alors**

Debut

$tf(tij,di) = 1+\log(tfij)$;

$idf(tij) = \log(nb\_doc/dfij)$;

$w(tj,q) = tf(tj,q) * idf(tj)$

$wtij = tf(tij,di) * idf(tij)$;

$RSV(di, Q) \leftarrow RSV(di, Q) + (wtij * w(tj,q))$;

Fin

Ordonner les documents selon la pertinence ;

Fin

III.6.4 Module Profil :

Cette méthode permet de récupérer les données personnelles de l'utilisateur, ainsi que ses préférences en recherche (centre d'intérêt et format de document préférés à retourner).

Méthode *insert_profil ( )*

Entrées : Données personnelles (nom, prénom, login , mot de passe, centre d'intérêt) , Types de documents préférés (pdf, html, doc) ;

Début

Ouvrir la connexion a la base de données ;

setData (données personnelles) ;

// Insèrer les données personnelles à la table_utilisateurs

setData (types de documents préférés) ;

// Insèrer les préférences à la table_preferences

Fermer la connexion à la base de données :

Fin

III.7 Conclusion

Ce chapitre a porté sur la conception de notre SRI personnalisé. Nous avons donné l'architecture globale, puis décrire ses différents modules et expliquer comment nous avons intégré le profil utilisateur dans les différentes phases du processus de recherche. A la fin , nous avons donné les algorithmes correspondant à notre raisonnement. Le chapitre suivant portera sur la mise en œuvre de ces algorithmes.

Chapitre IV : Mise En œuvre De System De RI Personnalisé

IV.1. Introduction

Ce dernier chapitre termine notre étude théorique en proposant une mise en œuvre des algorithmes que nous avons proposés dans le chapitre précédent. Nous donnons en premier l'architecture physique du SRI personnalisé, puis nous décrirons l'environnement dans lequel nous l'avons développé. Nous terminons avec des illustrations pour montrer le fonctionnement de notre système.

IV.2. Architecture physique du SRI personnalisé

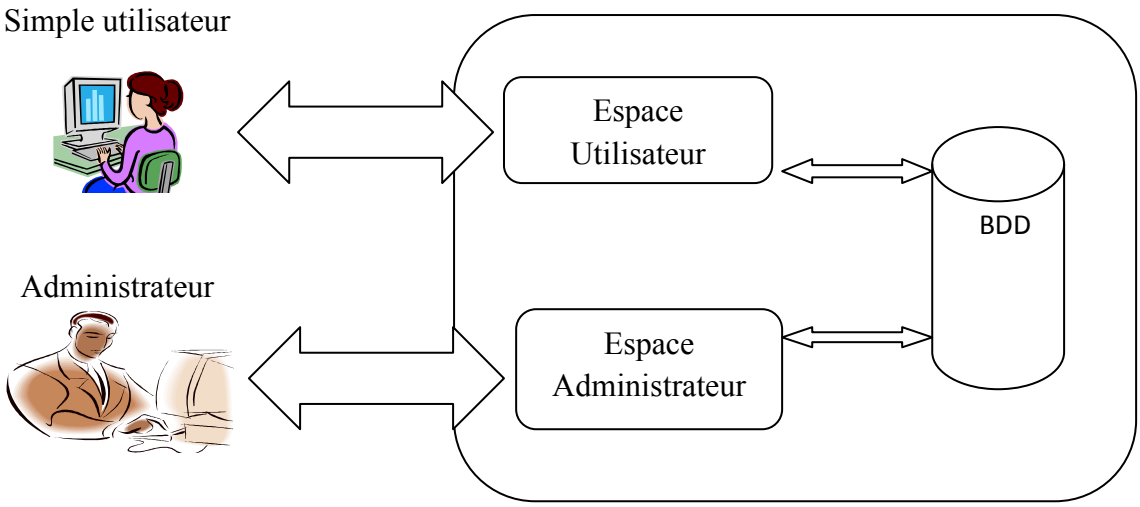

Figure IV.1 : Architecture physique du SRIP

IV.3. L'environnement de développement

Pour les besoin de mise en œuvre, nous avons opté de développer sur une plateforme. Windows ayant comme caractéristiques (Intel (R) Dual- Core 2, CPU à 2.20 Hz et 2Go de RAM) et le langage Java (JDK 1.6 Update 23), argumenté du fait de son succès et son essor en matière de développement d'application, notamment pour sa portabilité.

Pour le choix technique des outils, nous avons choisi :

- L'environnement NetBeans 6.7.1 pour développer le SRIP.

NetBeans est à l'origine un environnement de développement intégré (EDI) pour Java,placé en open source par Sun en juin 2000 sous licence CDDL et GPLv2 (Common Developmentand Distribution License). En plus de Java, NetBeans permet également desupporter différents autres langages, comme Python, C, C++, XML, Ruby, PHP et HTML. Ilcomprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multilangage,refactoring, éditeur graphique d'interfaces et de pages Web).Conçu en Java,

NetBeans est disponible sous Windows, Linux, Solaris (sur x86 etSPARC), Mac OS X et Open VMS.NetBeans permet de programmer et concevoir les interfaces utilisateur de manière visuelle.Pour ce faire, il offre de nombreux outils de conception visuelle qui permettent de concevoirles interfaces utilisateur avec rapidité et efficacité en attachant des événements et en modifiantles dispositions.

Pour notre réalisation nous avons utilisés la version 6.7.1 (NetBeans 6.7.1).

- WampServer pour construire la base de données.

Wampserver est une plateforme de développement web sous Windows, ilpermet de développer des applications web dynamiques à l'aide du serveur apache 2 comme serveur web, du langage de Script PHP comme interface graphique pourmanipuler les BD et MYSQL serveur de BD. C'est un paquetage contenant à la foisdeux serveurs (Apache et MYSQL), un Interpréteur de script (PHP), les deux interfacesSQL (PHP myadmin et SQL it manager) pour gérer automatiquement et facilementune plate forme permettant l'exploitation d'un site web en PHP qui éventuellementaurait besoin d'un accès à une base de données.

IV.4. Configuration du SRIP

Pour faire fonctionner notre SRIP, on suit les étapes suivantes :

1- Construction de la base de données via Wampserver :

La figure suivante présente la base de données suivant le schéma donné dans le chapitre précédant.

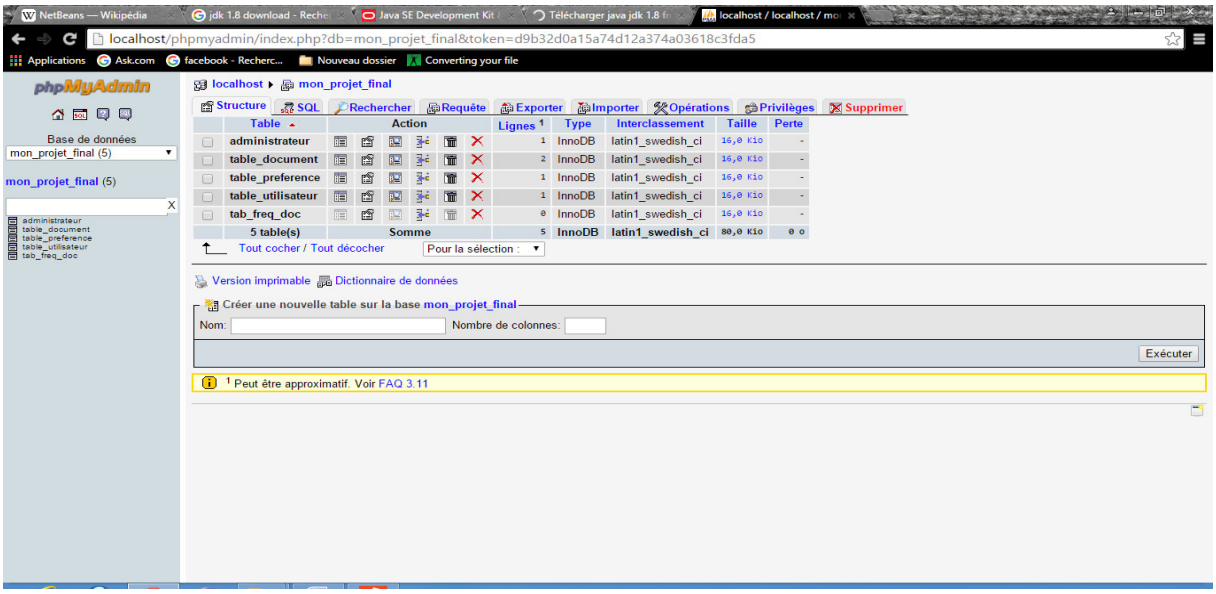

Figure IV.2 : Base de données via Wampserver

2- Déploiement des applications :

- Pour l'application SRIP, il suffit de lancer l'exécutable.

IV.5. Illustrations et Fonctionnement du SRIP

Nous allons maintenant présenter les principales fonctionnalités de notre système à travers ses interfaces graphiques. Commençons par celle de l'Interface principale.

Figure IV.3: Interface principale

IV.5.1. Espace Administrateur

L'Authentification

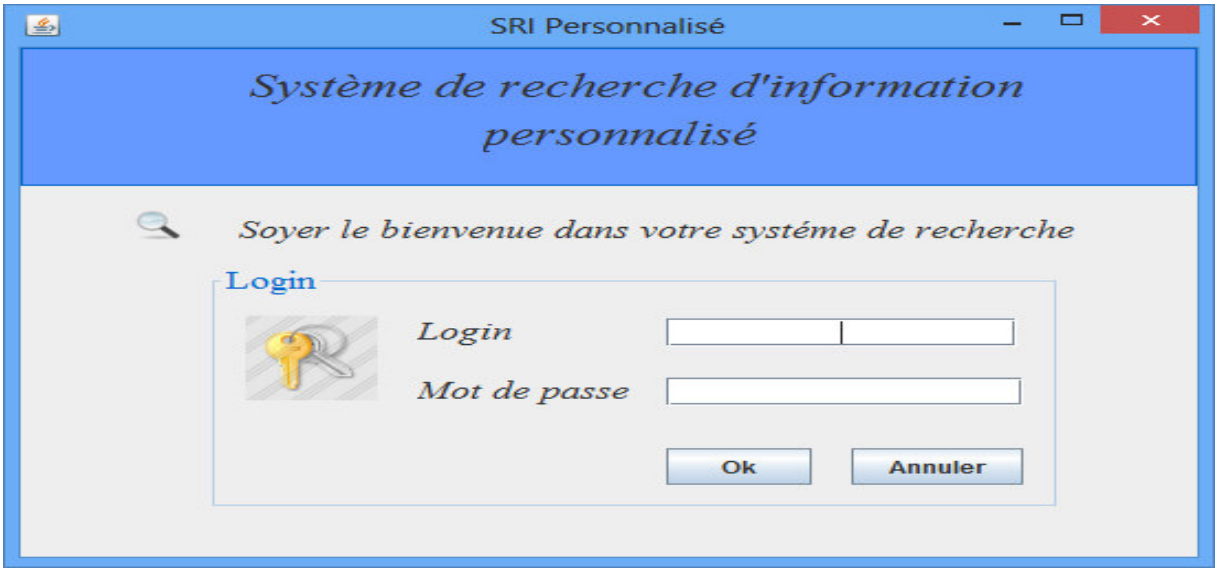

Figure IV.4: Espace d'authentification « Administrateur »

Le SRIP assure un minimum de sécurité via l'authentification par Login et Mot de passe. Une fois validé, l'administrateur a accès à son espace de gestion.

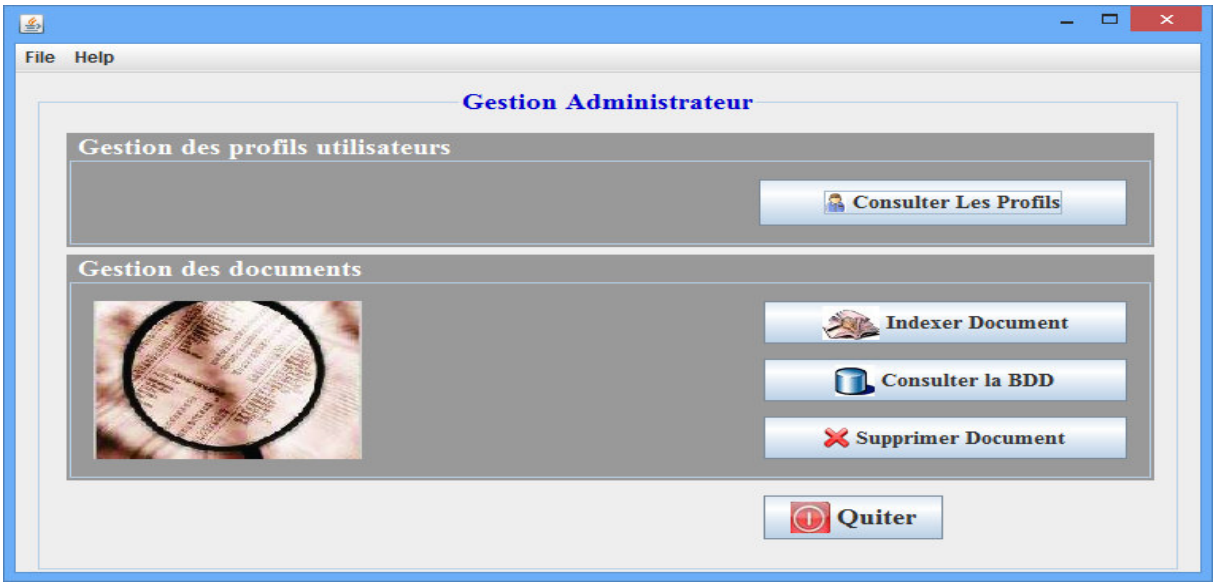

Figure IV.5: Interface de gestion Administrateur

Espace de Gestion Administrateur :

L'Administrateur peut assurer les tâches suivantes :

Consulter Les Profils : Permet de consulter le profil utilisateur. L'Administrateur s'intéresse au centre d'intérêt et aux préférences des utilisateurs. Grace à ceci, il pourra apporter une meilleure catégorisation aux documents.

Indexer Document : permet d'indexer un document et le catégorisé.

Figure IV.6: Gestion des Documents

Consulter La BDD : Permet à l'Administrateur de consulter la base de données.

Supprimer Document : Permet la suppression d'un document de la base de données.

IV.5.2. Espace Utilisateur :

Pour accéder à l'espace utilisateur, il faut s'authentifie.

Figure IV.7 : Interface d'authentification Utilisateur

Après authentification, le système nous renvoie l'interface de recherche suivante

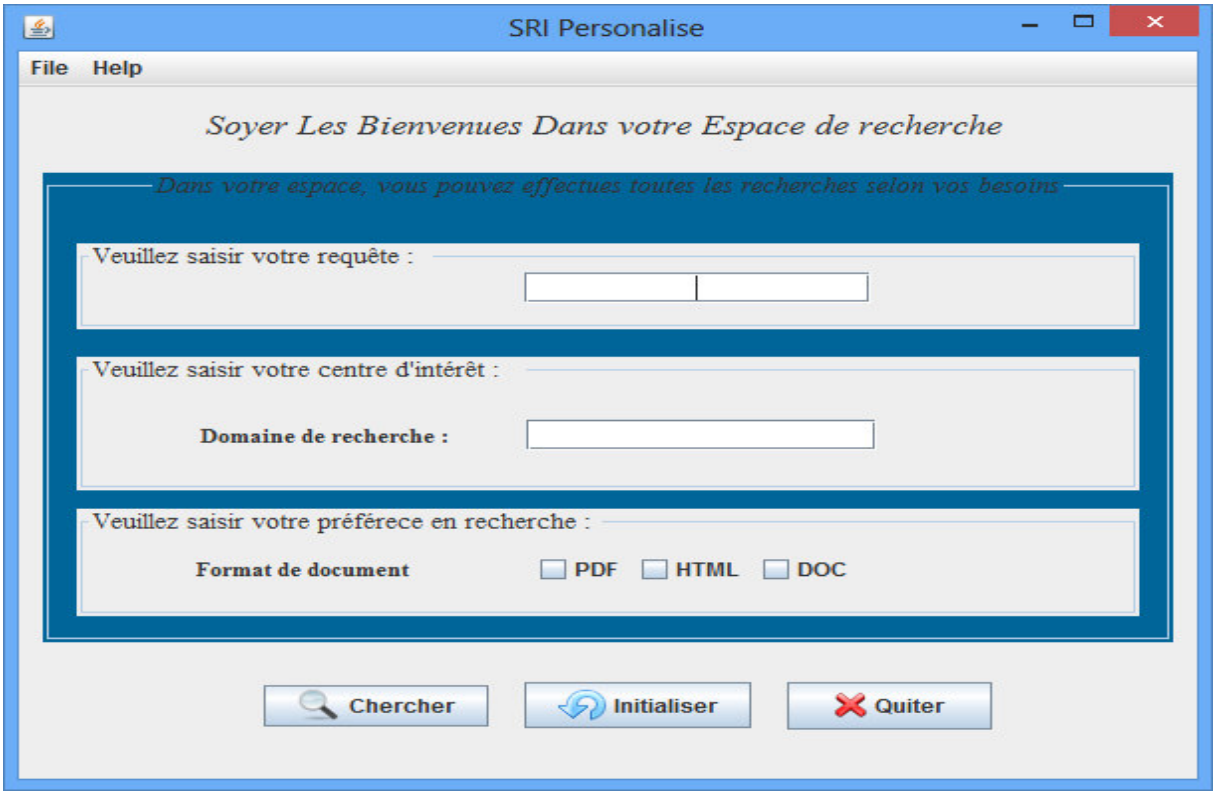

Figure IV.8 : Interface de recherche

En validant notre requête, le système retourne les résultats dans une nouvelle fenêtre.

L'inscription des utilisateurs :

Dans le cas ou l'utilisateur ne possède pas un compte sur le système, il peut en avoir un en remplissant le formulaire suivant :

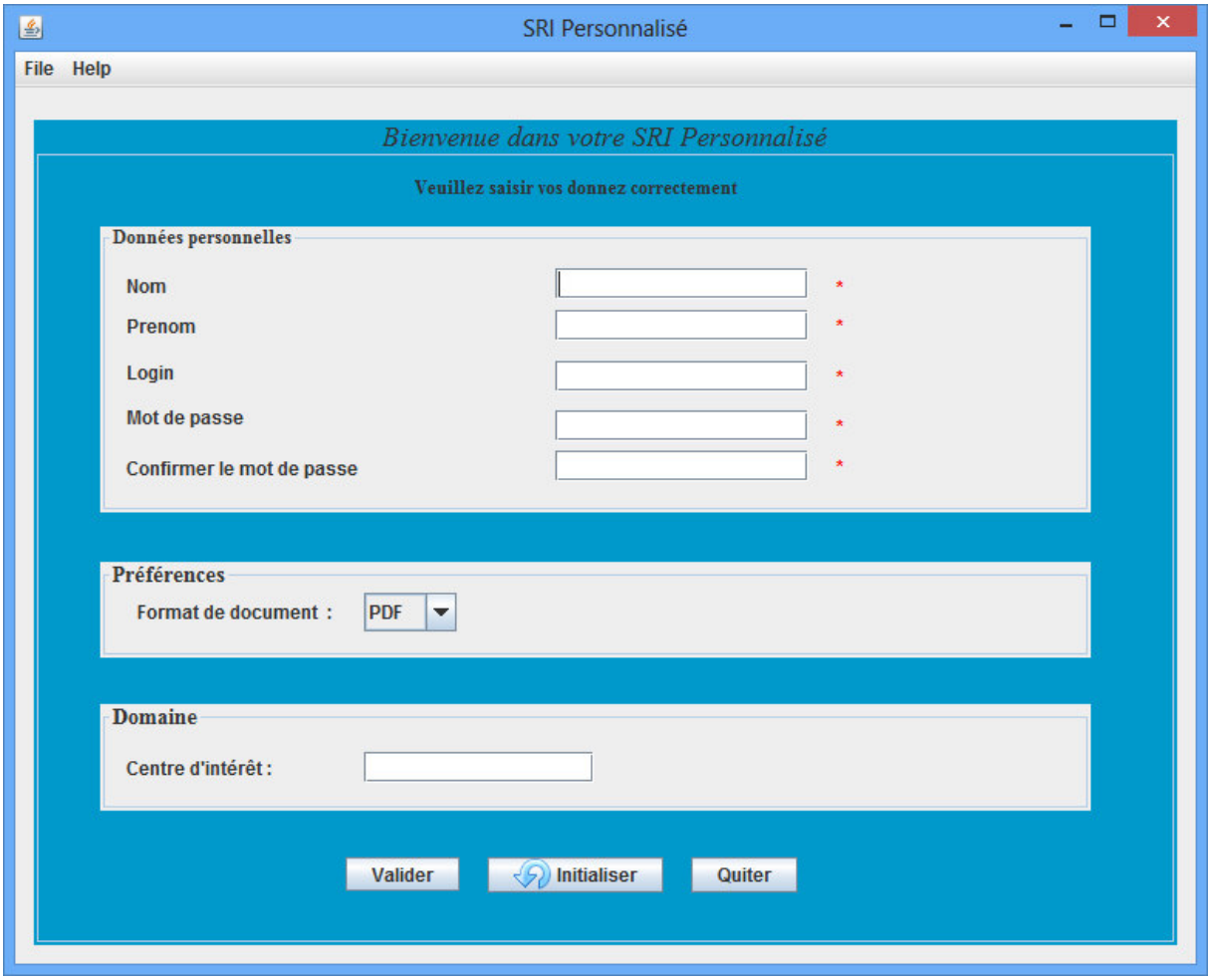

Figure IV.9 : Interface d'inscription Utilisateur

IV.6 Conclusion

Nous avons exposé dans ce chapitre une brève présentation du SRIP que nous avons implémenté. Nous avons opté pour une interface Administrateur de type application qui est consulté seulement par l'administrateur et une interface web d'interrogation qui permet de faire des recherches personnalisées et d'accéder au SRIP.

Conclusion Générale

Le travail présenté dans ce mémoire rentre dans le cadre de la recherche d'information personnalisée (RIP). Nous avons alors implémenté un modèle d'appariement pour une RI personnalisée permettant un accès personnalisé à l'information pertinente en fonction de chaque utilisateur.

Nous avons tout d'abord abordé la notion de la RI et rappeler quelques concepts de la RI Classique, puis traiter la Recherche d'Information personnalisée .En particulier, nous avons consacré un grand espace pour la modélisation personnalisée en se basant sur le profil utilisateur. Notre SRIP est modélisé sur la base d'un modèle vectoriel où nous avons intégrer le profil utilisateur dans la phase d'appariement du processus de recherche.

Le profil est représenté par deux unités informationnelles, à savoir les données personnelles et le centre d'intérêt. Grace auquel nous avons pu faire d'une part une indexation profilée des documents et d'autre part construire une requête profilée qui spécifie au mieux le besoin en information de l'utilisateur. Les résultats retournés sont ordonnés selon leurs degrés de pertinence.

Les différents algorithmes ressortis des étapes précédentes ont été mis en œuvre et donnent un SRI Personnalisé ayant une interface Administrateur et une Interface personnalisée destinée aux utilisateurs.

En perspective, nous envisageons, dans un premier temps, d'améliorer le calcul de pertinence des documents en utilisant d'autres modèles de RI, entre autres le modèle probabiliste.

[Perenon 00] : Pascal perenon, réalisation d'un prototype système de recherche spécifique Labo_RECODOC.UCBL.2000

[Amato 99] : G.amato, U.stroccia, user profil modeling and application to digital librarie.1999

[Amadieu 09] : Amadieu, effets of prior knowledye deversity of learning.2009

[P.borlund 03] ; P.borlund a fromework for evaluation of interactive information retrieval systems, information research, 2003

[Pednault 00] : P.D pednault, representation is every thing_CACM.2000

[Salton 71] : G.salton, the SMART retieval system : experiments in automatic document processing.1971

[Saracevic 97] : P.saracevic, the stratfied model of information retrieval interaction.1997

[Ingwersen 96] : P.ingwersen. cognitive perspectives of information interaction.1996

[Pretchener 99] : Alexonder pretchnner.Ontalogy Based personalised search.1999

[kwri 99] : T.Kwri, agents in delivering personlised search.1999

[kostadinov 03] : D.kostadinov, la personnalisation de l'information, définition de modèle de profil.2003

[W.Zemirli 08] : W.N.Zemerli. Modèle d'accès personnalisé à l'information basée sur les diagrammes d'influence integrant un profil utilisateur évolutif. U.Toulouse III.2008

Mogellan : mogellan.excite.com

[I.Boussoid 05] : Iphum.boussoid_ personnalisation de l'information et gestion des profils utilisateurs : approche fondée sur les ontologies.2005

[Saltysiaka et crobtree 98] : S.J.saltysiaka, I.crobtree_ automatic tearning of user profiles to words the personnalisation of agent services.1998

[K.Al.Maksssoud 08] : karine.Al.makssoud , système d'accès personnalisé à l'information. INSA.LYON.2008

[Mariam09] : Accès personnalisé à l'information : approche basée sur l'utilisation d'un profil utilisateur sémantique dérivé d'une ontologie de domaine a travers l'historique de session de recherche.2009

[Mariam, lynda, mohand, bilal 08] : Mariam daoud, construction des profils utilisateur a base d'ontologie pour une recherche d'information personnalisée.COKIA.2008

[Shen, Ton, Zhai 05] X.shen, B.Ton et C.Zhai.implicite user modeling for personalised search.2005

[Gowon 03] J.Gowon. a multiple model approach to pesonnalised information acces.2003

[Sieg, Mobasher,Burke,Prabu,lytinen 04] A.sieg, B.mobasher, R.burke, G.prabu et S.lytinen. Using concepts hierachies to enhance user queries in web_based information retrieval.2004

[Salton 83] : extended boolean information retrieval system.1983

[Ribeiro-Neto] : Modern Information Retrieval. Pearson Education Ltd., Harlow, UK,

2nd edn, 2011

[Baziz M] : Indexation Conceptuelle Guidée Par Ontologie Pour La Recherche d'Information. Thèse de Doctorat en Informatique de l'Université Paul Sabatier de Toulouse.2005

[Borlund] : P. Measures of relative relevance and ranked halflife: performance indicators for interactive in Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval.1998

[Cleverdon] : Progress in documentation. Evaluation of information retrieval systems. Journal of Documentation.1970

[Das, A. & Jain, A]: Indexing the World Wide Web: The journey so far. IGI Global. 2012

[Dominich, S] : Mathematical Foundations of Information Retrieval. Kluwer Academic Publishers, Dordrecht,Boston, London.2001

[Fox, C] : Lexical analysis and stoplists, Frakes W B, Baeza-Yates R (eds) Prentice Hall, New jersey.1992

[Fuhr, N] : Information Retrieval - From Information Access to Contextual Retrieval. In M. Eibl, C. Wolf, and C. Womser-Hacker, editors, Designing Information Systems. Festschrift für Jürgen Krause. UVK Verlagsgesellschaft.2005

[Harter, S] : Psychological relevance and information science. Journal of the American Society for Information Science (JASIS).1992

[Jacquemin, C., Daille, B., Royanté, J., and Polanco, X] : In vitro evaluation of a program for machine-aided indexing. Inf. Process. Manage.2002

[Kraft, D. H. and BueIl, D] : Fuzzy sets and generalized Boolean retrieval systems. International Journal on Man-Machine Studies.1983

[Manning, D., Raghavan, P. And Schute, H] : Introduction to Information Retrieval. Cambridge University Press.2008

[Porter, M] : An algorithm for suffix stripping. Program.1980

[Robertson, S. E] : The Probability Ranking Principle in IR. Journal of Documentation.1977

[Salton, G] : The smart Retrieval System: Experiments in Automatic Document Processing. Prentice-Hall.1971.

[Salton, G., McGill, M] : Introduction to Modern Information Retrieval. McGraw-Hill Int. Book Co.1984

[Singhal, A., Salton, G., Mitra, M., Buckley, C] : Document length normalization. Information Processing and Management.1996

[M^{r} Hammache Arezki] : thèse de doctorat en informatique pour thème de la RI : un modèle de langue combinant mots simple et mot composés . UMMTO